

UNIVERSITÉ DE MONTRÉAL

DÉTERMINATION DE LA TAILLE DES EFFECTIFS ET AFFECTATION DES
SÉQUENCES DE REPOS DANS LES HORAIRES D'EMPLOYÉS DE COMPAGNIES
DE TRANSPORT PUBLIC

MARC GENDRON
DÉPARTEMENT DE MATHÉMATIQUES ET DE GÉNIE INDUSTRIEL
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(MATHÉMATIQUES APPLIQUÉES)
DÉCEMBRE 2012

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé :

DÉTERMINATION DE LA TAILLE DES EFFECTIFS ET AFFECTATION DES
SÉQUENCES DE REPOS DANS LES HORAIRES D'EMPLOYÉS DE COMPAGNIES
DE TRANSPORT PUBLIC

présenté par : GENDRON Marc

en vue de l'obtention du diplôme de : Maîtrise ès Sciences Appliquées

a été dûment accepté par le jury d'examen constitué de :

M. SOUMIS François, Ph. D., président

M. HERTZ Alain, Doct. ès Sc., membre et directeur de recherche

M. FLEURENT Charles, Ph. D., membre

REMERCIEMENTS

J'aimerais remercier GIRO Inc. qui m'a supporté tout au long de ma maîtrise. Plus particulièrement, j'aimerais remercier Charles Fleurent, Loïc Bodart, Réjean Lessard et Alain Dallaire pour leurs nombreux conseils. Les discussions que nous avons eues sur mon projet et sur la recherche opérationnelle en général m'ont été très utiles. En plus de son support, Giro m'a fourni un environnement de test et les données essentielles aux expérimentations faites durant mon projet.

Je dois aussi remercier mon directeur de recherche Alain Hertz, pour son support et ses nombreux conseils pour ma recherche et la rédaction de mon mémoire. Il m'a donné de nombreuses pistes de recherche pour résoudre certaines embûches ou pour simplifier mes modèles parfois très compliqués.

Finalement, j'aimerais remercier ma famille pour son support. Particulièrement Andrée-Anne qui a enduré les soirées et les fins de semaine consacrées aux études et à la recherche. Je la remercie de m'avoir encouragé dans les moments où j'en avais besoin.

RÉSUMÉ

Les compagnies de transport public construisent les horaires de leurs employés en plusieurs étapes. Une de ces étapes est la construction des séquences de repos. Dans un environnement où plusieurs conventions collectives régissent différents groupes d'employés, cette étape est très importante. En effet, c'est à cette étape que le nombre exact d'employés requis pour chaque groupe est déterminé. L'objectif de ce mémoire est de modéliser les problèmes de construction des séquences de repos et de détermination de la taille des effectifs dans des horaires cycliques dans un contexte où il y a plusieurs groupes d'employés. Quelques modélisations seront proposées et seront comparées à l'aide d'expérimentations sur des jeux de données. Ces jeux de données réelles représentent différentes compagnies de transport public dans le monde.

Dans le cadre de ce mémoire, deux modélisations seront présentées. La première modélisation est une modélisation 0-1 linéaire. La deuxième modélisation est une composition de plusieurs flots. Les deux modélisations permettront de trouver une solution optimale des problèmes dont la taille dépasse 450 employés. En fait, nous montrons qu'avec la deuxième modélisation, la complexité du problème ne dépend pas du nombre d'employés ni du nombre de tâches.

ABSTRACT

Public transit companies build schedules for their employees in multiple steps. One of those steps is the construction of the sequence of rest days. In an environment where multiple collective agreements govern multiple employee groups, this step is very important. In fact, it is at this step that the exact number of employees required for each group is determined. The objective of this master's thesis is to model the problems of constructing rest day sequences and of determining the size of the workforce in a context where there are multiple employee groups. A few models will be proposed and compared in experimentations on data sets. Those data sets represent multiple public transit companies around the world.

In this master's thesis, two models will be presented. The first one is 0-1 linear mathematical model. The second mathematical model is a composition of multiple flows. Both models will lead to finding an optimal solution for problems with a size exceeding 450 employees. In fact, we will show that with the second model, complexity is not dependent of the number of employees nor the number of tasks.

TABLE DES MATIÈRES

| | |
|--|-----|
| REMERCIEMENTS | iii |
| RÉSUMÉ | iv |
| ABSTRACT | v |
| TABLE DES MATIÈRES | vi |
| LISTE DES TABLEAUX | ix |
| LISTE DES FIGURES | x |
| CHAPITRE 1 INTRODUCTION | 1 |
| 1.1 Définitions et concepts de base | 3 |
| 1.1.1 Groupe d'employé | 3 |
| 1.1.2 Horaire cyclique | 4 |
| 1.1.3 Règles et contraintes | 5 |
| 1.1.4 Objectifs et coûts | 7 |
| 1.2 Objectifs de recherche | 8 |
| CHAPITRE 2 REVUE DE LITTÉRATURE | 9 |
| 2.1 Horaire cyclique | 12 |
| 2.2 Groupes d'employés | 13 |
| CHAPITRE 3 MODÉLISATION LINÉAIRE PAR VARIABLES BINAIRES | 14 |
| 3.1 Description | 14 |
| 3.2 Variables | 15 |
| 3.3 Objectif | 16 |
| 3.4 Contraintes | 17 |
| 3.4.1 Activation des cycles dans le roulement | 17 |
| 3.4.2 Couverture des tâches | 18 |
| 3.4.3 Borne sur le nombre de positions par groupe | 20 |
| 3.4.4 Borne sur le nombre de repos par semaine | 20 |
| 3.4.5 Borne sur le nombre de repos par position | 20 |
| 3.4.6 Borne sur le nombre de repos par groupe d'employés | 21 |

| | | |
|---|---|----|
| 3.4.7 | Jour de repos forcé | 21 |
| 3.4.8 | Minimum de configuration de repos | 21 |
| 3.4.9 | Minimum de repos consécutifs | 25 |
| 3.4.10 | Bornes sur les séquences de travail | 25 |
| 3.5 | Complexité | 26 |
| 3.6 | Discussion sur l'activation des positions | 27 |
| CHAPITRE 4 MODÉLISATION PAR DÉCOMPOSITIONS EN FLOTS | | 29 |
| 4.1 | Description | 29 |
| 4.2 | Flot de séquence | 30 |
| 4.2.1 | Bornes sur le nombre de repos par semaine | 32 |
| 4.2.2 | Bornes sur les séquences de travail | 32 |
| 4.2.3 | Minimum de repos consécutifs | 33 |
| 4.2.4 | Jours de repos forcé | 33 |
| 4.2.5 | Groupes d'employés et cycle | 34 |
| 4.2.6 | Bornes sur le nombre de positions par groupe | 35 |
| 4.2.7 | Bornes sur le nombre de repos par position | 35 |
| 4.3 | Flots secondaires | 37 |
| 4.3.1 | Couverture des tâches | 37 |
| 4.3.2 | Minimum de configuration de repos | 38 |
| 4.3.3 | Bornes sur le nombre de repos par groupe d'employés | 39 |
| 4.4 | Méthode de résolution | 40 |
| 4.4.1 | Méthode linéaire | 40 |
| 4.5 | Complexité | 41 |
| 4.6 | Discussion sur l'activation des positions | 41 |
| CHAPITRE 5 EXPÉRIMENTATIONS | | 43 |
| 5.1 | Données | 43 |
| 5.2 | Environnement de test | 48 |
| 5.3 | Résultats | 48 |
| 5.3.1 | Modélisation linéaire | 48 |
| 5.3.2 | Modélisation par les flots | 48 |
| 5.4 | Analyse | 49 |
| CHAPITRE 6 CONCLUSION | | 51 |
| 6.1 | Synthèse des travaux | 51 |
| 6.2 | Limitations des solutions proposées | 52 |

| | |
|-------------------------------------|----|
| 6.3 Améliorations futures | 52 |
| RÉFÉRENCES | 54 |

LISTE DES TABLEAUX

| | | |
|-------------|--------------------------------------|----|
| Tableau 3.1 | Notation | 15 |
| Tableau 3.2 | Légende de complexité | 26 |
| Tableau 3.3 | Complexité des variables | 27 |
| Tableau 3.4 | Complexité des contraintes | 27 |

LISTE DES FIGURES

| | | |
|-------------|---|----|
| Figure 1.1 | Grille horaire, une semaine | 2 |
| Figure 1.2 | Grille horaire, deux semaines | 2 |
| Figure 1.3 | Grille horaire | 4 |
| Figure 1.4 | Grille horaire d'employé, avec cycles déroulés | 5 |
| Figure 1.5 | Grille horaire d'employé, avec cycle non complet | 6 |
| Figure 3.1 | Modèle de flot : respect de la couverture des tâches | 18 |
| Figure 3.2 | Horaire avec journée suivante | 22 |
| Figure 4.1 | Modèle de flot à 1 semaine, avec 2 repos consécutifs | 29 |
| Figure 4.2 | Modèle de flot à 4 semaines | 30 |
| Figure 4.3 | Modèle de flot | 31 |
| Figure 4.4 | Modèle de flot à 4 semaines, avec multiples commencements | 34 |
| Figure 4.5 | Modèle de flot à 4 semaines, avec multiples commencements et bornes sur le nombre de positions | 35 |
| Figure 4.6 | Modèle de flot à 4 semaines, borne sur le nombre de repos par position | 36 |
| Figure 4.7 | Combinaison de deux flots avec des nœuds parallèles | 37 |
| Figure 4.8 | Couverture des tâches avec plusieurs flots | 38 |
| Figure 4.9 | Minimum de configuration de repos avec plusieurs flots | 39 |
| Figure 4.10 | Bornes sur le nombre de repos par groupe d'employés | 40 |

CHAPITRE 1

INTRODUCTION

Le domaine du transport public a grossi dans les dernières décennies. Il y a une augmentation du nombre d'utilisateurs et de l'offre de service des compagnies. Par exemple, à Montréal entre 2002 et 2010, il y a eu une augmentation de 27 % du service offert aux usagers (STM, 2011) et, entre 1970 et 2011, il y a eu une augmentation de 65 % du nombre de déplacements. Cette situation est observée partout dans le monde. Pour la recherche opérationnelle, il s'agit d'un domaine en croissance où plusieurs développements restent à venir.

La construction des horaires des employés fait partie du cœur des préoccupations des compagnies de transport public. Depuis longtemps, le processus est découpé en 4 étapes. La première consiste à construire l'horaire des véhicules. La deuxième étape consiste à prendre les horaires des véhicules et à les répartir dans des journées de travail. La troisième étape consiste à construire les séquences de travail et de repos des employés. La quatrième étape consiste à affecter ces séquences à des chauffeurs. Ces quatre étapes ont été séparées à cause de leur complexité.

L'étape de construction des séquences de travail et de repos est faite en deux sous-étapes. La première consiste à affecter les journées de repos et la deuxième consiste à affecter des tâches. Par le passé, il était permis de faire l'affectation des repos seulement sur une période d'une semaine à la fois. Au-delà de la semaine, le problème devenait trop gros. Une deuxième raison justifiait ce choix : une semaine était idéale pour le marché nord-américain. En effet, la majorité des compagnies de transport public nord-américaines fonctionnent par ancienneté pour l'affectation des séquences aux chauffeurs. Les chauffeurs avec la plus grande ancienneté choisissent la séquence de travail qu'ils préfèrent. En général, ces séquences de travail durent une semaine. Les chauffeurs travaillent avec des horaires identiques pendant plusieurs mois. La figure 1.1 montre l'horaire de quatre employés qui répéteront le même travail pendant plusieurs mois. Dans cette figure, chaque ligne représente un employé et chaque colonne représente une journée de la semaine. Les journées de repos sont représentées par la lettre R. Les autres cases représentent des journées de travail. Chaque journée de travail contient une tâche. Ces tâches sont représentées par un chiffre.

| Employé | Semaine 1 | | | | | | |
|---------|-----------|-----|-----|-----|-----|-----|-----|
| | Dim | Lun | Mar | Mer | Jeu | Ven | Sam |
| 1 | R | R | 1 | 1 | 1 | 2 | 2 |
| 2 | 1 | 1 | 2 | 2 | 2 | R | R |
| 3 | 2 | 2 | R | R | 3 | 3 | 3 |
| 4 | 3 | 3 | 3 | R | R | 1 | 1 |

Figure 1.1 Grille horaire, une semaine

De l'autre côté de l'océan, le marché européen aimerait pouvoir affecter les journées de repos et les journées de travail pour plusieurs semaines à la fois. Dans ce marché, les compagnies essaient de diviser équitablement le travail entre tous les employés. La méthode qui a été utilisée pour répondre à ce besoin est de construire un horaire sur plusieurs semaines. Chaque employé est affecté à une première semaine. La semaine suivante, l'employé fera le travail de l'employé suivant. L'employé travaillant la dernière semaine fera ensuite le travail de la première semaine. Les employés vont itérer à travers toutes les semaines. Ainsi, un employé qui reçoit un horaire moins intéressant ne travaille qu'une seule semaine sur cet horaire. La semaine suivante, il pourra travailler avec l'horaire du prochain employé. Cette solution fonctionne, mais n'est pas parfaite. Les outils actuels permettant de créer ce genre d'horaire ne tiennent pas compte de toutes les règles de la convention collective. Par exemple, les règles qui ont un impact sur plusieurs semaines ne sont pas prises en compte dans un mécanisme de distribution sur une semaine. À titre d'illustration, la règle suivante n'est typiquement pas considérée : il n'est pas permis de travailler plus de 6 jours consécutifs. La figure 1.2 montre un horaire cyclique sur deux semaines. Cela signifie que les employés travaillent deux semaines puis feront le travail du prochain employé durant les deux semaines suivantes. Avec une vision aussi courte, on ne peut pas intégrer des éléments de personnalisation des horaires. Les horaires ne tiennent pas compte des vacances ou des formations des employés. Avec un horizon plus long, il pourrait être possible d'intégrer ce genre d'information.

| Employé | Semaine 1 | | | | | | | Semaine 2 | | | | | | |
|---------|-----------|-----|-----|-----|-----|-----|-----|-----------|-----|-----|-----|-----|-----|-----|
| | Dim | Lun | Mar | Mer | Jeu | Ven | Sam | Dim | Lun | Mar | Mer | Jeu | Ven | Sam |
| 1 | R | R | 1 | 1 | 1 | 2 | 2 | 2 | R | R | 1 | 1 | 1 | R |
| 2 | 1 | 1 | 2 | 2 | 2 | R | R | 3 | 3 | 3 | R | R | 2 | 2 |
| 3 | 2 | 2 | R | R | 3 | 3 | 3 | R | 2 | 2 | 2 | 2 | R | R |
| 4 | 3 | 3 | 3 | R | R | 1 | 1 | 4 | 4 | R | R | 3 | 3 | 3 |

Figure 1.2 Grille horaire, deux semaines

À long terme, plusieurs compagnies du marché européen aimeraient pouvoir construire des horaires sur plusieurs semaines. Cela permettrait de prendre en considération les règles des conventions collectives qui ne sont actuellement pas prises en compte. De plus, cela permettrait d'être plus équitable lors de la distribution des journées de travail. La distribution sur plusieurs semaines permettrait aussi d'égaliser le travail pour un employé dans sa séquence de travail sur plusieurs semaines.

S'il est vrai que considérer un horizon plus long qu'une semaine a des avantages, la prise en compte de plusieurs groupes d'employés en même temps a également une valeur ajoutée. Le fait de considérer les groupes d'employés séparément ne permet pas d'obtenir la solution optimale dans certaines situations. Certaines tâches peuvent être faites par différents groupes d'employés. Si on divise le problème par groupe d'employés, il devient difficile de savoir dans quel groupe mettre une tâche pour obtenir la solution optimale globalement. En considérant les différents groupes d'employés en même temps, il est facile d'atteindre une solution de moindre coût.

Le problème de distribution des repos n'inclut pas la distribution des tâches. Cependant, lors de la distribution des repos, il faut tenir en compte des différentes tâches à effectuer pour s'assurer qu'il y a assez de jours de travail pour exécuter les différentes tâches.

Plan du mémoire

Dans un premier temps, nous donnerons une description du problème étudié. Ensuite, une revue de la littérature sera présentée. Puis, deux modélisations seront détaillées. Des résultats d'expérimentation seront présentés pour les deux modélisations. Finalement une conclusion sera faite.

1.1 Définitions et concepts de base

1.1.1 Groupe d'employé

Dans le problème, il y a un ensemble de groupes d'employés. Chacun des groupes d'employés possède des règles d'affectation qui dictent la construction des séquences de repos. Les règles d'affectation sont généralement différentes d'un groupe à l'autre. Ces règles définissent les séquences de repos possibles et impossibles. Pour les séquences possibles, les règles attribuent généralement un coût à ces séquences. La plupart des règles sont déterminées à partir des conventions collectives et des lois en vigueur. Chaque groupe d'employés peut avoir sa propre convention collective. Un exemple de loi serait : « un employé est limité à un maxi-

mum de 6 jours de travail consécutifs. » En plus, certaines règles peuvent définir la préférence des employés. Par exemple, un employé pourrait préférer avoir 3 jours de repos par semaine au lieu de 2 et pourrait désirer obtenir des tâches de plus longue durée durant ses journées de travail. Dans les compagnies de transport public, on rencontre généralement entre 4 et 20 groupes d'employés ayant des règles différentes.

En plus des règles, les groupes d'employés définissent les tâches que les employés du groupe peuvent faire. Il existe une relation « plusieurs à plusieurs » entre les tâches et les groupes d'employés. C'est-à-dire qu'un groupe d'employés a plusieurs tâches disponibles et chaque tâche peut être affectée à l'un ou l'autre des groupes d'employés. Par contre, une tâche ne peut être faite que par un employé durant une journée de travail. La tâche doit donc être affectée à un seul groupe de travail.

1.1.2 Horaire cyclique

Pour commencer, un horaire a une taille en nombre de semaine. En général, l'horaire peut avoir entre 1 et 4 semaines. Ensuite, un horaire est divisé en groupe d'employés. Dans la figure 1.3, chaque ligne représente une position. Une position représente l'horaire d'un employé durant un intervalle de temps égal à la taille de l'horaire. Chaque colonne représente une journée dans l'horaire. Dans la figure, la colonne #1 représente le dimanche de la semaine 1. La colonne #10 représente le mardi de la semaine 2.

| Groupe | Position | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|--------|----------|-----------|-----|-----|-----|-----|-----|-----|-----------|-----|-----|-----|-----|-----|-----|
| | | Semaine 1 | | | | | | | Semaine 2 | | | | | | |
| | | Dim | Lun | Mar | Mer | Jeu | Ven | Sam | Dim | Lun | Mar | Mer | Jeu | Ven | Sam |
| A | 1 | | | | | | | | | | | | | | |
| A | 2 | | | | | | | | | | | | | | |
| A | 3 | | | | | | | | | | | | | | |
| A | 4 | | | | | | | | | | | | | | |
| B | 1 | | | | | | | | | | | | | | |
| B | 2 | | | | | | | | | | | | | | |
| B | 3 | | | | | | | | | | | | | | |
| C | 1 | | | | | | | | | | | | | | |
| C | 2 | | | | | | | | | | | | | | |
| D | 1 | | | | | | | | | | | | | | |
| D | 2 | | | | | | | | | | | | | | |
| D | 3 | | | | | | | | | | | | | | |
| E | 1 | | | | | | | | | | | | | | |
| E | 2 | | | | | | | | | | | | | | |

Figure 1.3 Grille horaire

Chaque groupe d'employés définit une taille de cycle. Cette taille donne le nombre de positions dans lesquelles les employés vont itérer. Cette taille de cycle peut valoir 1, ce qui signifie qu'un employé qui finit de travailler une position recommence au début de la même position. La taille de cycle peut aussi valoir 2. Dans ce cas, l'employé finissant la première position recommence au début de la position suivante. L'employé de la deuxième position recommence à la première position. La figure 1.4 montre l'horaire de 6 employés lorsqu'on déroule l'horaire dont la taille du cycle est de 3. La taille de cycle peut valoir n'importe quelle valeur entière.

| | | | | | | |
|-----------|------------|------------|------------|------------|------------|------------|
| Employé 1 | Position 1 | Position 2 | Position 3 | Position 1 | Position 2 | Position 3 |
| Employé 2 | Position 2 | Position 3 | Position 1 | Position 2 | Position 3 | Position 1 |
| Employé 3 | Position 3 | Position 1 | Position 2 | Position 3 | Position 1 | Position 2 |
| Employé 4 | Position 4 | Position 5 | Position 6 | Position 4 | Position 5 | Position 6 |
| Employé 5 | Position 5 | Position 6 | Position 4 | Position 5 | Position 6 | Position 4 |
| Employé 6 | Position 6 | Position 4 | Position 5 | Position 6 | Position 4 | Position 5 |

Figure 1.4 Grille horaire d'employé, avec cycles déroulés

Le nombre de positions dans un groupe d'employés est en général un multiple de la taille du cycle de ce groupe. Certaines compagnies pourraient cependant ne pas se limiter à un multiple du cycle. Dans ce cas, les dernières positions d'un groupe d'employés qui font partie d'un cycle incomplet itèrent entre eux. Dans l'exemple de la figure 1.5, le groupe d'employés possède 11 positions et a une taille de cycle de 3. Dans cet exemple, la position 10 et la position 11 font partie d'un cycle incomplet. Dans ce cas, l'employé finissant la position 10 recommencera à la position 11 et l'employé finissant à la position 11 recommencera à la position 10.

1.1.3 Règles et contraintes

Les conventions collectives et les préférences des employés donnent une série de règles qui doivent être respectées dans les groupes d'employés. Un échantillon représentatif en terme de complexité a été extrait de plusieurs compagnies de transport public. Cet échantillon permettra de modéliser ces règles en contraintes dans les différentes modélisations de ce document. En pratique dans les compagnies de transport public, ce ne sont pas toutes les règles de cet échantillon qui se retrouvent dans la même convention collective. Seul un sous ensemble de règles se retrouve dans une même convention collective. Voici la description de cet échantillon :

| Groupe | Position | Semaine 1 | | | | | | |
|--------|----------|-----------|-----|-----|-----|-----|-----|-----|
| | | Dim | Lun | Mar | Mer | Jeu | Ven | Sam |
| A | 1 | | | | | | | |
| A | 2 | | | | | | | |
| A | 3 | | | | | | | |
| A | 4 | | | | | | | |
| A | 5 | | | | | | | |
| A | 6 | | | | | | | |
| A | 7 | | | | | | | |
| A | 8 | | | | | | | |
| A | 9 | | | | | | | |
| A | 10 | | | | | | | |
| A | 11 | | | | | | | |

Figure 1.5 Grille horaire d'employé, avec cycle non complet

1. Il doit y avoir un nombre minimum et maximum de repos par semaine dans chaque position. Par exemple, la règle pourrait définir un minimum de 2 jours de repos et un maximum de 3 jours de repos par semaine.
2. Il doit y avoir un nombre minimum de repos consécutifs par semaine dans chaque position. Par exemple la règle pourrait définir un minimum de 2 jours de repos consécutifs par semaine.
3. Il doit y avoir un nombre minimum et maximum de repos par position. Par exemple, la règle pourrait définir un minimum de 9 jours de repos et un maximum de 10 jours de repos par position.
4. Il doit y avoir un nombre minimum et maximum de repos par groupe d'employés. Par exemple, la règle pourrait définir un minimum de 30 jours de repos et un maximum de 40 jours de repos par groupe d'employés par semaine.
5. Lorsqu'un jour de travail suit un jour de repos, il doit y avoir un nombre minimum et un nombre maximum de jours de travail consécutifs. Par exemple, un employé doit faire au minimum 4 jours de travail consécutifs, mais jamais plus de 6.
6. Les compagnies de transport public peuvent promettre à leurs employés qu'il y aura un minimum d'employés dans un groupe qui auront une fin de semaine de repos. Une fin de semaine de repos signifie d'avoir le samedi et le dimanche de repos.
7. Elles peuvent aussi promettre qu'il y aura un nombre minimum de semaines avec un double repos dans le groupe. Une semaine avec un double repos est une semaine où il

y a au moins deux jours de repos consécutifs.

8. Elles peuvent aussi promettre qu'il y aura un nombre minimum de semaines avec un triple repos dans le groupe. Une semaine avec un triple repos est une semaine où il y a au moins trois jours de repos consécutifs.
9. Elles peuvent finalement promettre un repos un jour particulier à tous les employés d'un groupe. Par exemple, tous les employés d'un groupe doivent avoir un repos le dimanche.

En plus des règles imposées par les conventions collectives et les préférences des employés, certaines contraintes peuvent être ajoutées. La principale contrainte qui peut être ajoutée est un nombre minimum et maximum de positions dans un groupe d'employés.

1.1.4 Objectifs et coûts

Toutes les règles énumérées à la section précédente peuvent être souples ou strictes. Une règle stricte est une règle qui ne peut pas être violée. Les règles souples peuvent ne pas être respectées en ajoutant une pénalité dans la fonction objectif. La pénalité est proportionnelle à l'écart de la valeur avec la règle. Prenons par exemple une règle souple imposant un minimum de deux repos par semaine avec une pénalité de 10. Une semaine avec exactement deux repos n'aurait aucune pénalité. Une semaine avec un repos aurait une pénalité de 10 et une semaine sans semaine de repos aurait une pénalité de 20.

La fonction objectif du problème est en deux parties. La première correspond à la somme des pénalités des règles souples non respectées. La deuxième est associée à la minimisation des effectifs excédentaires et à l'étalement sur tous les jours. Les effectifs excédentaires sont les employés qui travaillent une journée sans avoir de tâches affectées. Cela arrive lorsqu'il y a plus d'employés qui travaillent que de tâches à affecter. Pour minimiser le nombre d'effectifs supplémentaires, un terme sera ajouté à la fonction objectif. Pour un planificateur, il est plus facile d'éliminer une position lorsqu'il y a plusieurs colonnes avec des effectifs excédentaires que lorsque tous les effectifs excédentaires sont la même journée. C'est pourquoi il faut tenir en compte qu'un horaire ayant deux jours avec un effectif excédentaire est meilleur qu'un horaire ayant une journée avec deux effectifs excédentaires.

Les deux parties de la fonction objectif sont additionnées. Le coût associé à chaque règle et le coût associé aux effectifs excédentaires sont configurables par l'utilisateur de l'algorithme. Cela permet à l'utilisateur de spécifier ses priorités entre les règles et les effectifs excédentaires.

1.2 Objectifs de recherche

Le problème d'affectation des séquences de repos consiste à affecter les séquences de repos à des positions de sorte qu'il soit possible d'affecter le travail en respectant les règles des différents groupes d'employés. L'affectation doit se faire en minimisant le coût des séquences de travail et en minimisant le nombre d'employés en surplus. De plus, pour qu'il soit possible d'affecter les tâches, il faudra déterminer le nombre de positions dans chaque groupe.

L'objectif de cette recherche est de comparer plusieurs modélisations du problème d'affectation des séquences de repos et de détermination de la taille des effectifs dans un horaire cyclique où il y a plusieurs groupes d'employés. Tout cela se fera dans le but de trouver une modélisation efficace pour la résolution de ce problème.

CHAPITRE 2

REVUE DE LITTÉRATURE

Dans la littérature, la construction des séquences de repos et de jours de travail est une partie du problème général de construction d'horaires d'employés connue sous le nom anglais de « rostering » qui pourrait être traduit en français par roulement. Il y a deux parties dans la construction des roulements. La première partie consiste à affecter les séquences de repos et de jours de travail. La deuxième partie est l'affectation des tâches.

L'article (Ernst *et al.*, 2004) fait une revue complète du problème de construction d'horaires et de roulements. Dans l'article, la construction des horaires et des roulements est divisée en 6 étapes. Les deux étapes du précédent paragraphe font partie des 6 étapes de l'article. Ces étapes sont clairement définies. L'article donne les principaux domaines où la construction de roulements est appliquée. Finalement, l'article présente une énumération des différentes méthodes utilisées pour résoudre chacune des étapes. Chaque méthode est associée à une série d'articles décrivant l'application de cette méthode.

D'autres articles font une revue complète pour un domaine en particulier. C'est le cas de l'article (Arabeyre *et al.*, 1969) qui traite du problème de construction d'horaire et de roulement dans le domaine du transport aérien. Dans cet article, le problème de roulement est nommé « Linkage ». L'article (Nurmi *et al.*, 2011) traite du problème de roulement dans les compagnies de transport en autobus.

Plusieurs documents dans la littérature traitent séparément les deux parties du problème de création de roulements. C'est le cas de la thèse de doctorat de Bennett (1967) intitulée *Optimization of bus crew rosters : an application of combinatorial mathematics*. Cette thèse est l'un des premiers documents à traiter du problème de roulement. Dans cette thèse, Bennett utilise une routine de construction pour les séquences de jours de repos et de jours de travail. Cette routine permet de maximiser les journées de repos consécutifs tout en ayant les jours de repos distribués le plus également possible. En résumé, cette routine calcule le nombre de périodes de 4 repos consécutifs. La routine calcule aussi le nombre de périodes de 3 repos consécutifs et le nombre de périodes de 2 repos consécutifs. La routine distribue le plus équitablement possible toutes ces séquences de repos entre les employés. Ensuite, elle distribue les repos isolés restants.

L'article (Day et Ryan, 1997) décrit aussi une méthode en deux étapes pour la construction des roulements pour les hôtesses de l'air. La méthode utilisée est basée sur un horaire de 14 jours. Au début, toutes les séquences possibles de repos et de travail sont générées pour les 14 jours. Le nombre de séquences possibles n'est pas très élevé, car les règles sont très strictes. Il doit y avoir 5 jours de repos et 9 jours de travail sur 14 jours. Selon les tests effectués, il est possible de générer entre 1 et 400 séquences pour chaque employée. Dans les tests, l'article montre que pour 80 employées, 6832 séquences ont été générées. Une fois les séquences de repos et de travail générées, une séquence est attribuée à chaque employée. Pour faire cela, l'article propose d'utiliser un modèle mathématique de partitionnement. Ce modèle permet de choisir les séquences pour toutes les employées tout en s'assurant d'avoir assez d'employées pour couvrir les tâches.

D'autres articles proposent des méthodes de constructions des roulements en une seule étape. C'est le cas de l'article (Caprara *et al.*, 1998) qui affecte les séquences de repos en même temps que les tâches. L'affectation du travail est simplifiée par le fait que les auteurs considèrent que chaque tâche doit être faite chaque jour. Ce qui implique qu'une journée comme le dimanche aura autant de service que le lundi. Selon les auteurs, une simple heuristique peut balancer cette simplification après la construction du roulement. Le roulement est construit par une heuristique. Cette heuristique est en fait une méthode constructive dont les choix sont basés sur un modèle linéaire avec relaxation lagrangienne. Cette méthode heuristique a remporté la compétition FARO (Ferrovie Airo Rostering Optimization) en 1995.

L'article (Cappanera et Gallo, 2004) utilise un multiflot pour résoudre le problème de roulement. Dans le modèle de flot, chaque arc représente une tâche ou un repos donné à un employé spécifique. Les arcs sont placés entre les tâches ou les jours de repos de façon à ce qu'aucune règle de compatibilité entre les tâches ne soit brisée. Des contraintes linéaires sont ajoutées pour s'assurer qu'une tâche est affectée à une seule personne. D'autres contraintes et variables linéaires sont introduites pour s'assurer de respecter certaines règles dans certaines périodes, comme le maximum de jours de travail consécutifs.

Les articles (Gamache et Soumis, 1998) et (Gamache *et al.*, 1999) proposent deux méthodes par génération de colonnes pour la construction des roulements. Les deux méthodes se ressemblent. Dans la méthode proposée dans (Gamache *et al.*, 1999), le problème maître est constitué du choix d'un horaire pour chaque employé et le sous-problème est constitué de la construction d'un horaire pour un employé. Cette technique a l'avantage de pouvoir s'assurer de la validité des horaires dans le sous-problème. Le problème maître s'assure sim-

plement qu'il y a assez d'employés pour chaque tâche. Dans le sous-problème, la construction de l'horaire d'un employé se fait avec un problème de plus court chemin avec ressources limitées. Les contraintes qui sont sur un intervalle de temps (par exemple « minimum d'un jour de repos par semaine ») sont modélisées à l'aide d'une ressource. Plus il y a de ressources dans le problème de plus court chemin, plus la résolution peut être longue. Pour résoudre les gros problèmes avec plusieurs centaines d'employés, la technique de génération de colonnes nécessite plusieurs heures de calcul et n'atteint pas nécessairement la solution optimale.

Une des principales différences entre les articles (Gamache et Soumis, 1998) et (Gamache *et al.*, 1999) est le sous-problème. Dans le premier article, le sous-problème considère toutes les possibilités d'horaires, ce qui fait un trop gros réseau dans le cas des grosses compagnies aériennes. Dans le deuxième article, un réseau est toujours utilisé, mais il n'a pas la même forme. Ce qui permet de réduire considérablement la taille du réseau.

D'autres articles font un mélange de la construction en une étape et de la construction en deux étapes. En réalité, il y a toujours deux étapes. Cependant, durant la première étape, en plus d'affecter les journées de repos, les types de tâches sont affectés. Un type de tâche est une caractéristique principale de la tâche. Il peut s'agir de n'importe quelle caractéristique, mais les « plages horaires » sont souvent utilisées. Une « plage horaire » est la caractéristique qui indique si la tâche se fait le matin, l'après-midi ou le soir. Dans la deuxième étape, les types de tâches sont remplacés par les tâches. C'est exactement ce qui est fait dans l'article (Sodhi et Norris, 2004). La méthode utilisée dans cet article pour la première étape est d'énumérer tous les patrons d'affectation d'une semaine. Chaque patron définit les jours de repos et le type de tâche pour chaque jour de travail. Les règles d'affectation présentées dans l'article permettent de réduire considérablement le nombre de patrons. Chaque patron est alors représenté par un nœud dans un modèle de flots. Les patrons sont liés ensemble de façon à ce que la compatibilité entre les patrons soit respectée. Des contraintes linéaires sont ajoutées pour s'assurer d'avoir assez de chaque type de tâches pour chaque jour. Ce modèle de flot avec contraintes linéaire est ensuite résolu par un MILP.

L'article (Hartog *et al.*, 2009) propose une méthode très similaire à la méthode de l'article (Sodhi et Norris, 2004). Le traitement se fait aussi en deux étapes. En plus de permettre d'affecter les types de tâche dans la première étape, des types de repos sont aussi affectés. Le modèle utilisé pour résoudre le problème d'affectation des jours de repos est un modèle linéaire d'affectation. Chaque type de repos et type de tâches sont représentés par une variable binaire associée à chaque jour de chaque position. Ainsi, il y a des contraintes linéaires s'assurant

qu'il y a exactement une affectation choisie pour chaque jour et exactement le bon nombre d'affectations est choisi pour chaque type de tâche et chaque type de repos. Une autre variable binaire est associée à des patrons d'affectation non désirés par les employés. Chaque patron a un coût qui lui est associé. Si un patron se retrouve dans les affectations choisies, le coût du patron se retrouve dans le coût de l'affectation.

2.1 Horaire cyclique

Dans la littérature, plusieurs articles traitent du problème de roulement avec des horaires en continu. C'est le cas dans la plupart des compagnies de transport aérien comme dans les articles (Gamache et Soumis, 1998) et (Gamache *et al.*, 1999). En effet, dans ce domaine, la géographie est à prendre en compte. Prenons par exemple un employé qui a une formation à Montréal à une certaine date. On ne peut pas lui affecter un vol Montréal-Paris la veille de sa formation, car il ne sera pas de retour à Montréal à temps pour sa formation. La localisation des employés est donc importante. Ce n'est pas le cas des compagnies de transport urbain. En effet, un conducteur d'autobus commence toujours sa journée à un dépôt et termine sa journée à ce même dépôt. Avec la contrainte de géographie en moins, il est possible de faire des horaires cycliques anonymes. Plusieurs compagnies fournissent des horaires cycliques, car leurs employés aiment la régularité d'un horaire cyclique tout en ayant une certaine diversité de travail qu'un horaire à plusieurs semaines permet d'avoir.

Les premiers articles à traiter des horaires cycliques sont Bennett (1967) et (Bennett et Potts, 1968). Dans ces articles, un seul cycle est construit pour tous les employés. Le cycle est composé d'une semaine par employé et le nombre d'employés est déterminé avant l'affectation des repos. L'algorithme permet de traiter un seul cycle. Les contraintes cycliques sont directement intégrées à la routine de construction des horaires.

L'article (Sodhi et Norris, 2004) et l'article (Hartog *et al.*, 2009) traitent aussi d'horaires cycliques. Dans le premier les contraintes cycliques sont modélisées sous forme de flot. Dans le deuxième, les contraintes sont modélisées dans un modèle linéaire.

Finalement, l'article (Caprara *et al.*, 1998) propose aussi un modèle linéaire où la faisabilité d'un horaire cyclique est vérifiée par un modèle de flot. Ce modèle produit des cycles d'un horaire où chaque cycle devra effectuer toutes les tâches avant la fin du cycle.

2.2 Groupes d'employés

Dans la littérature, peu d'articles traitent plusieurs groupes d'employés simultanément. Plusieurs articles mentionnent que l'approche est limitée à des groupes d'employés indépendants. C'est le cas des articles Bennett (1967), (Bennett et Potts, 1968), (Sodhi et Norris, 2004) et (Hartog *et al.*, 2009). Dans ces articles, les méthodes utilisées sont limitées à un groupe d'employés à la fois. Les méthodes proposées par Bennett (1967) et (Bennett et Potts, 1968) ne pourraient pas facilement être adaptées parce que ce sont des méthodes constructives. Il faudrait déterminer une logique de construction qui pourrait ne plus ressembler aux méthodes originales. Les méthodes des articles (Sodhi et Norris, 2004) et (Hartog *et al.*, 2009) pourraient être adaptées en ajoutant une contrainte linéaire. Cependant, il n'est pas facile de prédire le comportement du solveur linéaire après cet ajout.

Bien que les articles (Gamache et Soumis, 1998) et (Gamache *et al.*, 1999) ne traitent pas du sujet de plusieurs groupes d'employés, il pourrait être possible d'adapter le modèle du sous-problème pour s'assurer qu'une tâche est affectée à un employé seulement s'il est dans un groupe qui peut exécuter cette tâche.

CHAPITRE 3

MODÉLISATION LINÉAIRE PAR VARIABLES BINAIRES

3.1 Description

Il y a plusieurs façons de modéliser linéairement le problème d'affectation des repos. Celle qui sera présentée dans ce chapitre permet de modéliser les différents types de règles possibles dans les problèmes d'affectation des séquences de repos. Dans cette modélisation, les jours de repos sont représentés par une variable binaire qui indiquera si une position à une journée donnée est en repos.

Puisqu'il faut déterminer la taille des groupes d'employés, le nombre de positions dans chaque groupe n'est pas connu d'avance. Pour permettre de fixer les variables binaires décrites plus haut, il faut avoir un nombre de positions fixées. Pour faire cela, un nombre de positions suffisamment grand pour couvrir toutes les tâches sera utilisé. Ces positions pourront être activées ou désactivées pour avoir le bon nombre de positions dans chaque groupe d'employés.

Il serait possible d'avoir une variable d'activation pour chaque position. Elle permettrait d'écrire les contraintes linéaires en conséquence. Cependant, pour simplifier l'écriture du modèle mathématique, les positions seront activées par cycle. Cela permettra d'écrire plus simplement les contraintes cycliques. Une discussion sur l'activation des positions sera faite en fin de ce chapitre.

Tout au long de ce chapitre, une notation sera utilisée. Un résumé de cette notation est fait dans le tableau 3.1. Cette notation sera présentée dans le texte au cours de ce chapitre.

Tableau 3.1 Notation

| | |
|-----------|---|
| R | Ensemble des groupes d'employés. |
| P | Ensemble des positions dans tous les groupes d'employés. |
| P_r | Ensemble des positions du groupe $r \in R$. |
| C | Ensemble des colonnes. |
| D_c | Ensemble des tâches à affecter dans la colonne $c \in C$. |
| D_{rc} | Ensemble des tâches à affecter dans la colonne $c \in C$ pouvant être affecté au groupe $r \in R$. |
| R_d | Ensemble des groupes d'employés dans lesquels la tâche $d \in D$ peut être affectée. |
| J | Ensemble des jours de semaine (Dimanche, Lundi, Mardi, etc.). |
| j_c | Le jour de semaine $j \in J$ correspondant à la colonne $c \in C$. |
| W | Ensemble des semaines |
| c_{jw} | La colonne $c \in C$ correspondant au jour $j \in J$ de la semaine $w \in W$. |
| $Cycle_r$ | La taille des cycles du groupe $r \in R$. |
| S | Ensemble des cycles. |
| S_r | Ensemble des cycles du groupe $r \in R$. |
| P_s | Ensemble des positions du cycle $s \in S$. |
| s_p | Le cycle correspondant à la position p . |

3.2 Variables

Pour commencer, considérons P l'ensemble des positions. Chacun des éléments de P représente une position dans un des groupes d'employés. Soit C l'ensemble des colonnes. Chacun des éléments de C représente une journée, c'est-à-dire une colonne dans le tableau 1.3. Dans le modèle mathématique, les variables booléennes X_{pc} représenteront les repos pour une position $p \in P$ et une colonne $c \in C$:

$$X_{pc} = \begin{cases} 1 & \text{Si la position } p \in P \text{ a un repos dans la colonne } c \in C. \\ 0 & \text{Sinon.} \end{cases}$$

Soit l'ensemble des cycles S . Chaque position est associée à un cycle. Notons s_p , le cycle de la position p et notons P_s , l'ensemble de toutes les positions du cycle s . Dans le modèle, les variables booléennes Y_s représenteront l'activation d'un cycle donné :

$$Y_s = \begin{cases} 1 & \text{Si le cycle } s \in S \text{ est actif.} \\ 0 & \text{Sinon.} \end{cases}$$

Ces deux types de variables constituent la base du modèle de ce chapitre. D'autres variables seront définies dans la section sur les contraintes. Elles permettront d'écrire les contraintes.

3.3 Objectif

L'objectif est en deux parties. La première partie est la somme de toutes les pénalités associées aux règles souples qui n'ont pas été respectées. Les pénalités exactes dépendent des conventions collectives et des préférences des employés. Plus de détails sur ces pénalités seront donnés dans le chapitre 5. La deuxième partie est un ensemble de pénalités associées aux effectifs excédentaires. Ces pénalités devront faire en sorte que les effectifs excédentaires soient répartis sur toutes les colonnes. Dans la formule de la fonction objectif ci-dessous, la première partie représente la somme des pénalités associées aux règles et aux préférences. Le deuxième terme représente les pénalités reliées aux effectifs excédentaires.

$$\min(\sum_v e_v * V_v + \sum_{i=1}^7 U^i O_{ic}) \quad (3.1)$$

Pour toute contrainte de forme $ax \leq b$ représentant une règle stricte (par exemple, l'équation 3.12), il est possible de transformer cette contrainte pour obtenir une règle souple. La contrainte devient $ax + e_1 = b$ avec $e_1 \geq 0$. La variable d'écart e_1 est ajoutée à la première partie de la fonction objectif en multipliant la pénalité associée à la règle. Cette pénalité est représenté par la constante V_v .

$$\sum_v e_v * V_v \quad (3.2)$$

Pour calculer les pénalités associées aux effectifs excédentaires, définissons premièrement quelques variables. Soit O_c , une variable représentant le nombre d'employés en trop à la colonne $c \in C$. Voici la contrainte qui s'assure que la variable vaut exactement le nombre d'employés en surplus.

$$\sum_{p \in P} (Y_{sp} - X_{pc}) - |D_c| = O_c \quad \forall c \in C \quad (3.3)$$

Soit O_{ic} , une variable binaire pour $1 \leq i \leq 6$. Soit O_{ic} , une variable entière avec $i = 7$. Ces variables serviront à donner des coûts exponentiels au nombre d'effectifs excédentaires. Voici la contrainte liant ces variables à la variable O_c .

$$\sum_{i=1}^7 O_{ic} = O_c \quad \forall c \in C \quad (3.4)$$

La formule 3.5 indique le terme ajouté à la fonction objectif pour répartir les effectifs excédentaires. Ce terme fait en sorte que les variables O_{ic} avec un petit i seront utilisées

avant les variables avec un grand i . Dans la formule, la constante U est à la puissance i . Contrairement aux autres formules le symbole i n'est pas un indice, mais une puissance. Ainsi deux employés en surplus auront tendance à être sur deux journées différentes, car il en coûte U fois plus cher d'avoir deux employés en surplus la même journée que d'avoir deux employés en surplus sur deux journées. La constante U est donnée par l'utilisateur pour contrôler l'importance de réduire les effectifs excédentaires par rapport aux pénalités des règles et des préférences.

$$\sum_{i=1}^7 U^i O_{ic} \quad (3.5)$$

3.4 Contraintes

3.4.1 Activation des cycles dans le roulement

Pour commencer, on s'assure que les cycles actifs d'un groupe d'employés se retrouvent au début du groupe d'employés et les cycles inactifs se retrouvent à la fin.

$$Y_s \leq Y_{s-1} \quad \forall r \in R, s \in S_r \quad (3.6)$$

Pour faciliter l'écriture des prochaines contraintes, on s'assure qu'il y a des repos seulement dans les positions des cycles actifs.

$$\sum_{c \in C} X_{pc} \leq |C| Y_{s_p} \quad \forall p \in P \quad (3.7)$$

Tel que défini dans les sections 3.4.4 et 3.4.5, UOS_r est le nombre maximum de jours de repos par semaine pour le groupe $r \in R$, et UOP_r est le nombre maximum de jours de repos par position dans le groupe d'employés $r \in R$. Pour resserrer la contrainte 3.7, on peut bien entendu remplacer $|C|$ par $|W|UOS_r$ ou par UOP_r .

Dans les prochaines contraintes, les variables X serviront à identifier les journées de repos. Elles serviront aussi à identifier les journées de travail. Une journée de repos peut être présente seulement dans les cycles actifs. Par contre, dans les cycles inactifs, il n'y a aucun jour de repos. Les variables X avec une valeur égale à 0 représenteront des journées de travail dans les cycles actifs. Elles représenteront aussi toutes les journées dans les cycles inactifs. Les journées dans les cycles inactifs ne sont ni en repos ni en travail. Lorsque nécessaire, les prochaines contraintes considéreront les variables Y pour différencier les journées de travail dans les cycles actifs et les journées dans un cycle inactif.

Jusqu'à maintenant, dans un cycle actif, les journées de repos ne sont pas obligatoires. Toutes les journées dans les cycles actifs pourraient être des journées de travail. Par contre, cela ne fait aucun sens qu'un employé travaille toujours sans avoir de jour de repos. Les règles des conventions collectives doivent obliger un minimum de repos. Certaines des prochaines contraintes tiennent pour acquis qu'il y a au moins 1 repos par cycle. Les règles des conventions collectives devraient faire en sorte qu'il y ait au moins 1 repos. Par exemple, la contrainte 3.12 oblige un minimum de repos par semaine.

3.4.2 Couverture des tâches

Les contraintes ci-dessous permettent de respecter un nombre minimum de jours de travail dans les positions de chaque groupe d'employés. Ce minimum de jours de travail est déterminé à partir des tâches à affecter. Le modèle d'optimisation doit donner une solution d'affectation des repos qui permettra de couvrir toutes les tâches. Le but de la contrainte n'est pas d'affecter les tâches, mais de s'assurer qu'il sera possible de les affecter par la suite. Pour modéliser cette contrainte, un modèle de flot sera utilisé pour chaque colonne.

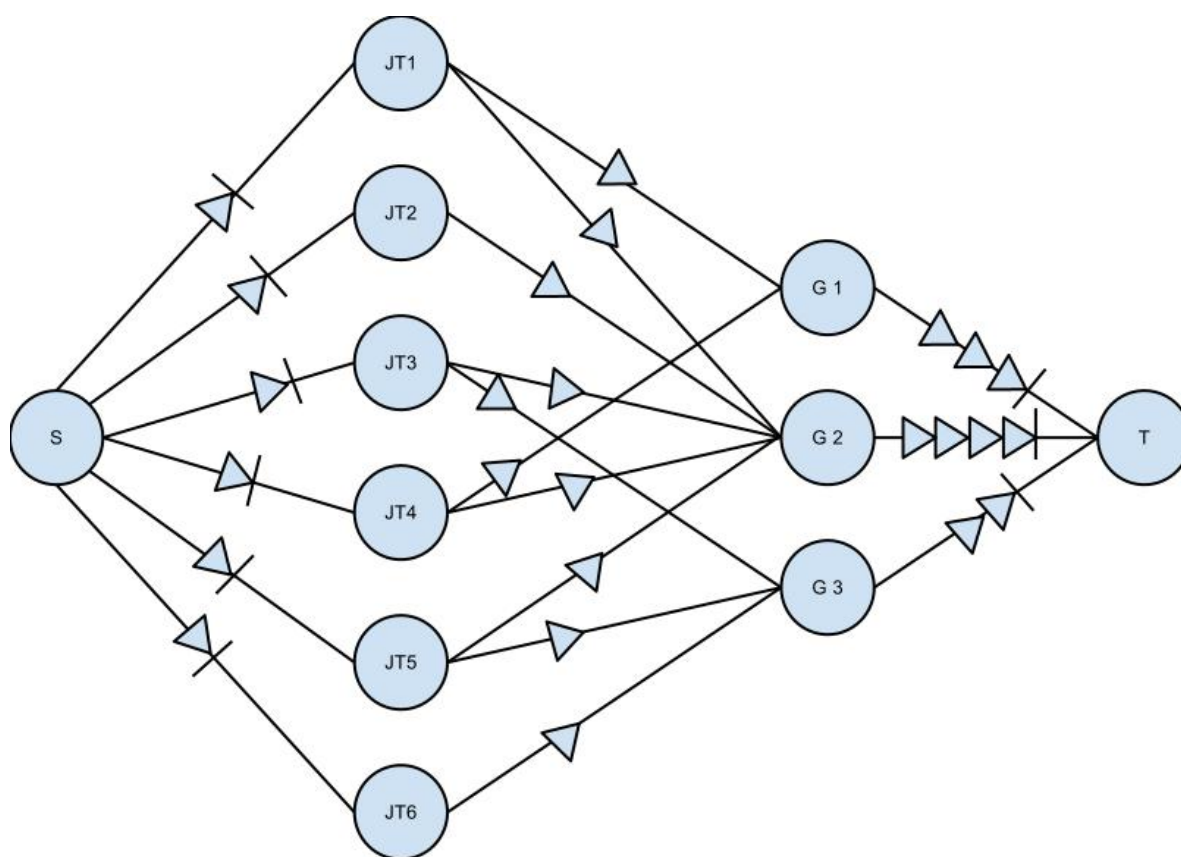


Figure 3.1 Modèle de flot : respect de la couverture des tâches

Dans la figure 3.1, le nœud S représente la source et le nœud T représente le puits. Chaque nœud de la colonne de gauche représente une tâche à affecter. Chaque nœud de la colonne de droite représente un groupe d'employés. Toutes les tâches de gauche ont un arc venant de la source avec un flot égal à 1. Il y a un arc entre une tâche et chaque groupe d'employés dans lequel la tâche peut être affectée. Finalement, il y a un arc entre chaque groupe d'employés et le puits avec un flot maximal correspondant au nombre de journées de travail dans les positions de ce groupe d'employés. Ce modèle de flot est construit pour chaque colonne de sorte que les tâches de chaque colonne ne peuvent pas être affectées à une autre colonne.

Variables

Voici de nouvelles variables qui permettront d'écrire le modèle de flot dans le modèle linéaire. Puisqu'un modèle de flot a toujours une solution optimale en nombres entiers quand les contraintes sont en nombres entiers, il est possible de mettre les variables du modèle de flot en nombres réels. Considérons D_c , l'ensemble des tâches à distribuer dans la colonne $c \in C$. Chaque tâche $d \in D_c$ peut être affectée à l'ensemble des groupes d'employés défini par R_d . À l'inverse, l'ensemble D_{rc} contient toutes les tâches qui peuvent être affectées au groupe d'employés $r \in R$ dans la colonne $c \in C$. Chacun des arcs entre la source et la tâche doit valoir exactement 1. Pour faciliter l'écriture du modèle de flot, la constante S_{cd} représente la quantité de flots passant sur un arc entre la source et une tâche d de la journée c . Cette constante vaut exactement 1. La variable DR_{cdr} représente la quantité de flots passant sur un arc entre une tâche $d \in D_c$ et un groupe d'employés $r \in R$. La variable T_{cr} représente la quantité de flots passant sur un arc entre un groupe d'employés $r \in R$ et le puits.

Contraintes

Voici la contrainte de conservation de flot des nœuds de tâches.

$$S_{cd} = \sum_{r \in R_d} DR_{cdr} \quad \forall c \in C, d \in D_c \quad (3.8)$$

Voici la contrainte de conservation de flot des nœuds de groupe d'employés.

$$T_{cr} = \sum_{d \in D_{rc}} DR_{cdr} \quad \forall c \in C, r \in R \quad (3.9)$$

Voici la contrainte qui impose un flot maximal sur les arcs entre les groupes d'employés et le puits. L'ensemble P_r contient toutes les positions de tous les cycles du groupe d'employés r .

$$T_{cr} \leq \sum_{p \in P_r} Y_{sp} - X_{pc} \quad \forall c \in C, r \in R \quad (3.10)$$

3.4.3 Borne sur le nombre de positions par groupe

À chaque groupe, il peut y avoir une contrainte qui impose minimum et un maximum de positions. La contrainte sur le maximum de positions est directement intégrée dans les ensembles P_r et S_r . Il y aura au plus le nombre de positions maximal de la contrainte dans l'ensemble P_r . Dans l'ensemble S_r , il y aura au maximum un nombre de cycles qui ne permettra pas de dépasser le nombre maximal de positions de la contrainte. La contrainte ci-dessous permet d'avoir un minimum de positions actives dans le groupe d'employés. Soit L_r , le minimum de positions actives dans le groupe r .

$$\sum_{s \in S_r} Y_s Cycle_r \geq L_r \quad \forall r \in R \quad (3.11)$$

3.4.4 Borne sur le nombre de repos par semaine

Il peut y avoir une contrainte qui impose un minimum de jours de repos par semaine pour chaque position d'un groupe. Soit LOS_r , le nombre minimum de jours de repos par semaine dans le groupe d'employés r . Soit W , l'ensemble des semaines. Soit J , l'ensemble des jours de la semaine. Soit c_{wj} , la colonne c associée au jour j de la semaine w . Dans la contrainte ci-dessous, le membre de droite vaut le minimum seulement si le cycle de la position est actif. Si le cycle est inactif, le membre de droite vaut 0.

$$\sum_{j \in J} X_{pc_{wj}} \geq LOS_r Y_{s_p} \quad \forall r \in R, p \in P_r, w \in W \quad (3.12)$$

Il peut y avoir aussi une contrainte qui impose un maximum de jours de repos par semaine de chaque position d'un groupe. Soit UOS_r , le nombre maximum de jours de repos par semaine pour le groupe $r \in R$. La contrainte sur le maximum de jours de repos peut être transformée en minimum de jours de travail par semaine. Dans la contrainte ci-dessous, le membre de droite représente le minimum de jours de travail si le cycle de la position est actif. Sinon, il vaut 0.

$$\sum_{j \in J} (1 - X_{pc_{wj}}) \geq (7 - UOS_r) Y_{s_p} \quad \forall r \in R, p \in P_r, w \in W \quad (3.13)$$

3.4.5 Borne sur le nombre de repos par position

La contrainte suivante impose un minimum de jours de repos par position d'un groupe. Soit LOP_r , le nombre minimum de jours de repos par position dans le groupe d'employés $r \in R$. Dans la contrainte ci-dessous, le membre de droite vaut le minimum seulement si le

cycle de la position est actif. Si le cycle est inactif, le membre de droite vaut 0.

$$\sum_{c \in C} X_{pc} \geq LOP_r Y_{s_p} \quad \forall r \in R, p \in P_r \quad (3.14)$$

Il peut aussi avoir une contrainte sur le nombre maximum de jours de repos par position d'un groupe. Soit UOP_r , le nombre maximum de jours de repos par position pour le groupe $r \in R$. Comme pour la contrainte sur le maximum du nombre de repos par semaine, la contrainte peut être transformée en minimum de jours de travail. Dans la contrainte ci-dessous, le membre de droite représente le minimum de jours de travail si le cycle de la position est actif. Sinon, il vaut 0.

$$\sum_{c \in C} (1 - X_{pc}) \geq (|C| - UOP_r) Y_{s_p} \quad \forall r \in R, p \in P_r \quad (3.15)$$

3.4.6 Borne sur le nombre de repos par groupe d'employés

On peut également imposer un nombre minimum de jours de repos par groupe d'employés. Soit LOG_r , le nombre minimum de jours de repos dans le groupe d'employés $r \in R$.

$$\sum_{c \in C} \sum_{p \in P_r} X_{pc} \geq LOG_r \quad \forall r \in R \quad (3.16)$$

Le même principe est appliqué pour le nombre maximum de jours de repos par groupe d'employés. Soit UOG_r , le nombre maximum de jours de repos dans le groupe d'employés $r \in R$.

$$\sum_{c \in C} \sum_{p \in P_r} X_{pc} \leq UOG_r \quad \forall r \in R \quad (3.17)$$

3.4.7 Jour de repos forcé

Un jour de la semaine peut être obligatoirement un jour de repos. Définissons RF_{rj} comme étant égale à 1 si la journée $j \in J$ doit être un jour de repos dans le groupe r , 0 sinon.

$$X_{pc_{wj}} \geq Y_{s_p} RF_{rj} \quad \forall r \in R, p \in P_r, w \in W, j \in J \quad (3.18)$$

3.4.8 Minimum de configuration de repos

Variables

Les règles sur les minimums de doubles et triples repos par semaine sont difficiles à modéliser par les variables déjà introduites. En effet, il faut calculer le nombre de semaines avec

au moins deux ou trois journées de repos consécutifs. Il ne faut pas compter le nombre de double repos dans le groupe d'employés. En effet, il serait possible qu'un employé ait deux doubles repos dans la même semaine. Il ne faudrait pas qu'il compte pour deux employés ayant des doubles repos.

Définissons la journée suivante d'une journée « A » comme étant la journée « B » qui suit « A » dans le même cycle. Dans la figure 3.2, la taille du cycle du groupe d'employés est de 2. La journée qui suit la journée « A » est la journée « B ». La journée qui suit la journée « D » est la journée « E ». La journée qui suit la journée « F » est la journée « A ».

| Groupe | Position | Semaine 1 | | | | | | |
|--------|----------|-----------|-----|-----|-----|-----|-----|-----|
| | | Dim | Lun | Mar | Mer | Jeu | Ven | Sam |
| A | 1 | A | B | C | | | | D |
| A | 2 | E | | | | | | F |
| A | 3 | | | | | | | |
| A | 4 | | | | | | | |

Figure 3.2 Horaire avec journée suivante

Pour chaque positions p et chaque colonne c , considérons le couple (p,c) . Définissons $N^i(p,c)$ comme étant la fonction qui retourne un couple (p,c) représentant la $i^{ème}$ journée suivante. Par exemple, dans la figure 3.2, le couple représentant la journée « A » est $(1,1)$, le couple représentant la journée « B » est $(1,2)$ et le couple représentant la journée « C » est $(1,3)$. Le premier voisin de A est donné par $N^1(1,1)$ et vaut $(1,2) = B$. Le deuxième voisin de « A » est donné par $N^2(1,1)$ et vaut $(1,3) = C$.

Pour calculer le nombre de positions avec des doubles repos, il faut premièrement repérer les doubles repos. La variable I_{wp}^j servira à identifier le début des doubles repos. La variable vaudra 1 si et seulement si le jour $j \in J$ de la semaine $w \in W$ dans la position p est un repos et que la journée suivante est aussi un repos.

$$I_{wp}^j = \begin{cases} 1 & \text{Si la journée } j \text{ de la semaine } w \text{ de la position } p \text{ est un repos et} \\ & \text{si la journée suivante l'est aussi.} \\ 0 & \text{Sinon.} \end{cases}$$

Après avoir détecté tous les doubles repos, il faut détecter les semaines contenant au moins 1 double repos. La variable I_{wp} servira à identifier les semaines ayant au moins 1 double

repos.

$$I_{wp} = \begin{cases} 1 & \text{Si la position } p \text{ a au moins 2 journées de repos consécutifs pendant la semaine } w. \\ 0 & \text{Sinon.} \end{cases}$$

Le même principe est appliqué pour les triples repos. La variable L_{wp}^j permet d'identifier le commencement de trois journées de repos consécutifs.

$$L_{wp}^j = \begin{cases} 1 & \text{Si la journée } j \text{ de la semaine } w \text{ est un repos dans la position } p \text{ et que les} \\ & \text{deux journées suivantes sont aussi des repos.} \\ 0 & \text{Sinon.} \end{cases}$$

Après avoir détecté tous les triples repos, il faut détecter les semaines contenant au moins 1 triple repos. La variable L_{wp} servira à identifier les semaines ayant au moins 1 triple repos.

$$L_{wp} = \begin{cases} 1 & \text{Si la position } p \text{ a au moins 3 jours de repos consécutifs pendant la semaine } w. \\ 0 & \text{Sinon.} \end{cases}$$

Lien entre variables

Les deux prochaines contraintes permettent de s'assurer que la variable I_{wp}^j vaut 1 si la journée courante et la journée suivante sont des repos.

La contrainte ci-dessous s'assure que I_{wp}^j est égal ou inférieur à 0 si les deux journées ne sont pas des repos. Elle s'assure aussi que la variable est égale ou inférieure à 0.5 si l'une des deux journées est un repos. La variable ne peut valoir 1 que si les deux journées sont des repos.

$$I_{wp}^j \leq \frac{X_{pc_{jw}} + X_{N^1(p, c_{jw})}}{2} \quad p \in P_r, w \in W, r \in R, j \in J \quad (3.19)$$

La contrainte ci-dessous s'assure que la variable I_{wp}^j vaut au moins 1 si les deux journées sont des repos. Si au plus une journée est un repos, la variable doit être supérieure ou égale à 0. En combinant cette contrainte et celle ci-dessus, on s'assure que la variable vaut 1 si et seulement si les deux journées sont des repos.

$$I_{wp}^j \geq X_{pc_{jw}} + X_{N^1(p, c_{jw})} - 1 \quad p \in P_r, w \in W, r \in R, j \in J \quad (3.20)$$

Les deux prochaines contraintes permettent de s'assurer que la variable I_{wp} vaut 1 si et seulement s'il y a au moins 1 double repos durant la semaine w dans la position p .

La contrainte ci-dessous s'assure que la variable I_{wp} vaut au moins le nombre de $I_{wp}^j = 1$ divisé par le nombre de jours dans la semaine. Par exemple, si dans la semaine il y a une seule variable I_{wp}^j qui vaut 1, alors la variable I_{wp} devra être supérieure ou égale à $1/7$.

$$I_{wp} \geq \sum_{j \in J} I_{wp}^j / |J| \quad p \in P_r, w \in W, r \in R \quad (3.21)$$

La contrainte ci-dessous s'assure que la variable I_{wp} ne vaut pas plus que la somme de toutes les variables I_{wp}^j . Ainsi, si toutes les variables I_{wp}^j de la semaine valent 0, alors I_{wp} vaudra 0. En combinant cette contrainte avec celle ci-dessus, on s'assure que la variable I_{wp} vaut 1 si et seulement si au moins une des variables I_{wp}^j vaut 1.

$$I_{wp} \leq \sum_{j \in J} I_{wp}^j \quad p \in P_r, w \in W, r \in R \quad (3.22)$$

Le même principe que celui de la variable I_{wp}^j est appliqué à la variable L_{wp}^j .

$$L_{wp}^j \leq \frac{X_{pc_{jw}} + X_{N^1(p, c_{jw})} + X_{N^2(p, c_{jw})}}{3} \quad \forall p \in P_r, w \in W, r \in R, j \in J \quad (3.23)$$

$$L_{wp}^j \leq X_{pc_{jw}} + X_{N^1(p, c_{jw})} + X_{N^2(p, c_{jw})} - 2 \quad \forall p \in P_r, w \in W, r \in R, j \in J \quad (3.24)$$

Le même principe que celui de la variable I_{wp} est appliqué à la variable L_{wp} .

$$L_{wp} \geq \sum_{j \in J} L_{wp}^j / |J| \quad p \in P_r, w \in W, r \in R \quad (3.25)$$

$$L_{wp} \leq \sum_{j \in J} L_{wp}^j \quad p \in P_r, w \in W, r \in R \quad (3.26)$$

Contraintes

Il est maintenant possible d'écrire la contrainte associée à la règle du nombre minimum de fins de semaine de repos. Soit LW_r , le nombre minimum de fins de semaine de repos du groupe d'employés r .

$$\sum_{p \in P_r} I_{wp}^j \geq LW_r \quad \forall w \in W, r \in R, j = \text{samedi} \quad (3.27)$$

Soit LDO_r , nombre minimum de position dans le groupe d'employé r devant avoir au moins un double repos par semaine. Un double repos est une séquence d'au moins deux jours de

repos consécutifs. Voici la contrainte associée à la règle d'un nombre minimum de doubles repos par semaine :

$$\sum_{p \in P_r} I_{wp} \geq LDO_r \quad \forall w \in W, r \in R \quad (3.28)$$

Soit LTO_r , nombre minimum de position dans le groupe d'employé r devant avoir au moins un triple repos par semaine. Un triple repos est une séquence d'au moins trois jours de repos consécutifs. Voici la contrainte associée à la règle d'un nombre minimum de triples repos par semaine :

$$\sum_{p \in P_r} L_{wp} \geq LTO_r \quad \forall w \in W, r \in R \quad (3.29)$$

3.4.9 Minimum de repos consécutifs

Un minimum de deux ou trois repos consécutifs par semaine peut être demandé par les conventions collectives. Soit MD_r , une constante valant 1 s'il doit y avoir au moins deux repos consécutifs par semaine à chaque position du groupe d'employés $r \in R$. 0 sinon. Soit MT_r , une constante valant 1 s'il doit y avoir au moins deux repos consécutifs par semaine à chaque position du groupe d'employés $r \in R$. 0 sinon.

$$I_{wp} \geq MD_r Y_{s_p} \quad \forall w \in W, r \in R, p \in P_r \quad (3.30)$$

$$L_{wp} \geq MT_r Y_{s_p} \quad \forall w \in W, r \in R, p \in P_r \quad (3.31)$$

3.4.10 Bornes sur les séquences de travail

Soit LTA_r , le nombre minimum de jours de travail consécutifs qu'il doit y avoir à chaque position du groupe d'employés r . La contrainte ci-dessous permet d'avoir au minimum LTA_r jours de travail suivant la journée X_{pc} si la journée X_{pc} est une journée de repos et la journée $X_{N^1(p,c)}$ est une journée de travail.

$$\sum_{z=1}^{LTA_r} (Y_{s_p} - X_{N^z(p,c)}) \geq LTA_r (X_{pc} - X_{N^1(p,c)}) \quad \forall p \in P_r, c \in C, r \in R \quad (3.32)$$

Soit UTA_r , le nombre maximum de jours de travail consécutifs à chaque position du groupe d'employés r . La contrainte ci-dessous permet de s'assurer qu'il y a au moins 1 jour de repos dans les $UTA_r + 1$ jours suivants la journée X_{pc} si la journée X_{pc} est une journée de repos et

la journée $X_{N^1(p,c)}$ est une journée de travail.

$$\sum_{z=1}^{UT A_r + 1} (X_{N^z(p,c)}) \geq (X_{pc} - X_{N^1(p,c)}) \quad \forall p \in P_r, c \in C, r \in R \quad (3.33)$$

3.5 Complexité

Chaque groupe d'employés de chaque compagnie a des règles différentes pour les séquences de repos et de travail. Ce sont ces règles qui déterminent les contraintes et les variables à avoir. Il y a des contraintes et des variables communes à tous les groupes d'employés. Cependant, pour chaque groupe d'employés, la complexité va varier en fonction des contraintes et des variables nécessaires pour modéliser les contraintes du groupe. Au final, la complexité du modèle dépend des règles de chaque groupe d'employés de la compagnie.

En plus des variables de base et des variables facultatives, il y a des variables d'écart permettant de rendre une contrainte souple. Ces variables d'écart sont associées à une pénalité dans la fonction objectif.

Les principales grandeurs qui influencent la complexité sont indiquées dans le tableau 3.2. Chaque symbole de ce tableau représente une grandeur. Les symboles seront utilisés pour décrire la complexité des variables et des contraintes.

Tableau 3.2 Légende de complexité

| Symbole | Description | Exemple de grandeur |
|---------|--|-----------------------|
| P | le nombre de positions | 800 positions |
| C | le nombre de colonnes | 28 colonnes |
| S | le nombre de cycles | 200 cycles |
| D_c | le nombre de tâches à affecter dans la colonne c | 600 tâches $c \in C$ |
| R | le nombre de groupes d'employés | 10 groupes d'employés |
| W | le nombre de semaines | 4 semaines |

Le tableau 3.3 indique quelles sont les variables les plus usuelles (dites de base), et celles qui dépendent des conventions collectives (dites facultatives). Ce tableau montre également la complexité de chacune de ces variables en fonction du nombre de positions, du nombre de colonnes et d'autres informations.

Tableau 3.3 Complexité des variables

| | | |
|------------|-------------|--------------|
| X_{cp} | Base | $ C P $ |
| Y_s | Base | $ S $ |
| O_c^i | Base | $8 C $ |
| DR_{cdr} | Base | $ C D_c ^1$ |
| T_{cr} | Base | $ C R $ |
| I_{wp}^j | Facultative | $7 W P $ |
| I_{wp} | Facultative | $ W P $ |
| L_{wp}^j | Facultative | $7 W P $ |
| L_{wp} | Facultative | $ W P $ |

Finalement, le tableau 3.4 indique les contraintes de base (les plus usuelles) et les contraintes facultatives (dépendantes des conventions collectives). Chacune des contraintes facultatives peut être transformée en contrainte souple. Dans le cas d'une contrainte souple, la complexité de la contrainte est la même que la complexité des variables à ajouter pour rendre la contrainte souple.

Tableau 3.4 Complexité des contraintes

| | | |
|---|-------------|----------------------|
| Effectifs excédentaires | Base | $2 C $ |
| Activation de cycle | Base | $ S + P $ |
| Couverture des tâches | Base | $ C D_c + 2 C R $ |
| Borne sur le nombre de positions par groupe | Facultative | $ R $ |
| Borne sur le nombre de repos par semaine | Facultative | $2 P W $ |
| Borne sur le nombre de repos par position | Facultative | $2 P $ |
| Borne sur le nombre de repos par groupe | Facultative | $2 R $ |
| Jours de repos forcé | Facultative | $7 P W $ |
| Minimum de configurations de repos | Facultative | $32 P W + 3 W R $ |
| Minimum sur les repos consécutifs | Facultative | $2 P W $ |
| Bornes sur les séquences de travail | Facultative | $2 P C $ |

3.6 Discussion sur l'activation des positions

Dans la modélisation précédente, il a été mis en évidence que les contraintes étaient liées à l'activation des cycles et aux formes des cycles. Par exemple, dans la contrainte 3.32, la variable d'activation des cycles Y_s est utilisée pour s'assurer de compter des journées de travail

1. En général, les tâches peuvent être associées à un nombre très limité de groupe d'employés. Ce nombre est d'environ 1 ou 2. Alors la complexité de la variable DR_{cdr} n'est pas en fonction du nombre de groupe d'employés qu'il y a dans le problème.

et non pas des journées de cycles non actifs. La fonction $N^i(pc)$ est liée à la taille du cycle, car elle retourne toujours la bonne journée suivante, peu importe si elle se trouve dans la même position ou dans une autre position du même cycle. Alors il serait difficile d'activer les positions de façon indépendante.

L'activation des positions une à une serait utile dans le cas où l'activation d'un cycle inclut trop de positions et où la désactivation du cycle n'en inclut pas assez. Il peut y avoir seulement un cycle incomplet par groupe d'employés. Considérant cela, il est possible d'ajouter plusieurs cycles de tailles différentes au groupe d'employés. Ces cycles de taille différente permettront d'activer exactement le bon nombre de positions dans le groupe d'employés.

CHAPITRE 4

MODÉLISATION PAR DÉCOMPOSITIONS EN FLOTS

4.1 Description

Toutes les règles énumérées à la section 1.1.3 peuvent être modélisées par un problème de flot. En effet, les règles définissent les séquences de repos valides et invalides. Il peut aussi y avoir un coût associé aux séquences valides. Une unité de flot dans le modèle montre une séquence de travail/repos. Par exemple, dans la figure 4.1, le flot représenté par les flèches foncées montre quelle est la séquence de journées de travail et de journées de repos. Les nœuds de ce flot représentent soit une journée de repos (R), soit une journée de travail (T). Le nœud (S) représente la source et le nœud (P) représente le puits. Dans l'exemple, il y a deux journées de travail suivies de deux journées de repos suivies de trois journées de travail. Il est possible d'introduire les contraintes qui manipulent la forme du flot. Dans le même exemple, il aurait été impossible d'avoir les deux journées de repos non consécutives. En effet, les deux journées de repos doivent être consécutives, car lorsqu'on entre dans un nœud représentant un jour de repos, il n'est pas possible de se rendre à un autre jour de travail tant qu'on n'a pas passé par un deuxième nœud représentant un repos.

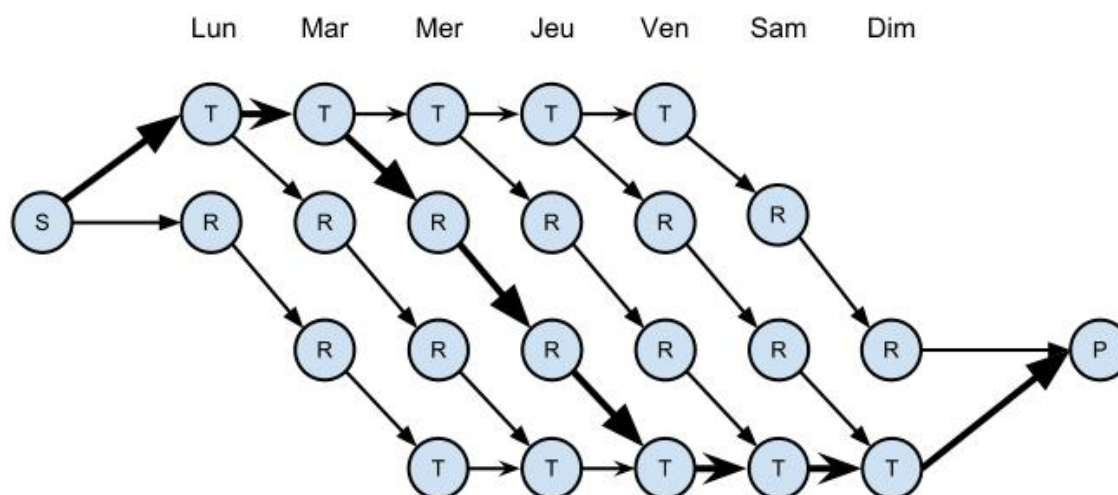


Figure 4.1 Modèle de flot à 1 semaine, avec 2 repos consécutifs

Un modèle de flot utilisé pour respecter les règles des séquences de journées de repos n'est pas suffisant pour modéliser toutes les contraintes du problème. En effet, il faut avoir suffisamment de journées de travail dans les séquences pour permettre d'affecter toutes les tâches. Un second modèle de flot permettra de gérer cette contrainte.

Dans ce chapitre, il sera démontré qu'il est possible de construire des modèles de flots pour faire respecter chaque règle. Ensuite, il y aura une discussion sur la combinaison des flots et les différentes méthodes de résolution.

4.2 Flot de séquence

À partir des règles de la section 1.1.3, il est possible de construire un flot qui permet de respecter presque toutes ces règles. La figure 4.2 montre un modèle de flot représentant un horaire de 4 semaines avec un cycle égal à 1. Chaque carré représente une semaine. Le carré est en fait une simplification d'un modèle de flot représentant une semaine. La figure 4.3 représente un agrandissement du carré pointillé.

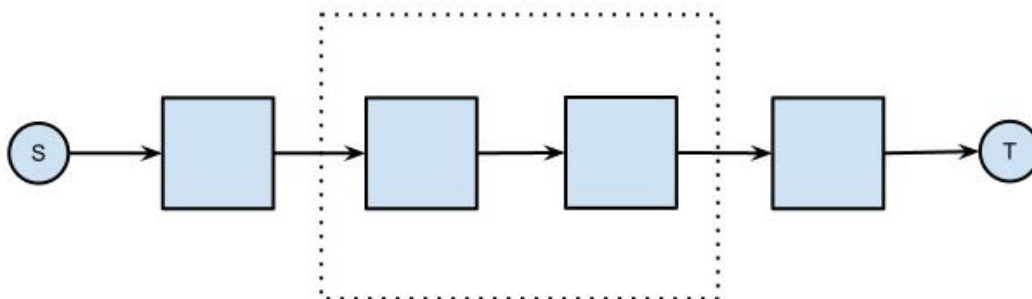


Figure 4.2 Modèle de flot à 4 semaines

La figure 4.3 représente 2 semaines adjacentes dans un même cycle. Chaque colonne de cette figure représente une journée. Les journées de 8 à 14 sont dans la première semaine. Les journées de 15 à 21 sont dans la deuxième semaine. En hauteur, la figure est séparée en trois parties par des lignes pointillées. La première partie représente une semaine ayant seulement des repos non consécutifs. La deuxième partie représente une semaine ayant au moins 2 repos consécutifs. La troisième partie représente une semaine ayant au moins trois repos consécutifs. Pour l'instant, nous ne donnerons pas plus de détails sur ces parties. Elles seront utiles un peu plus tard.

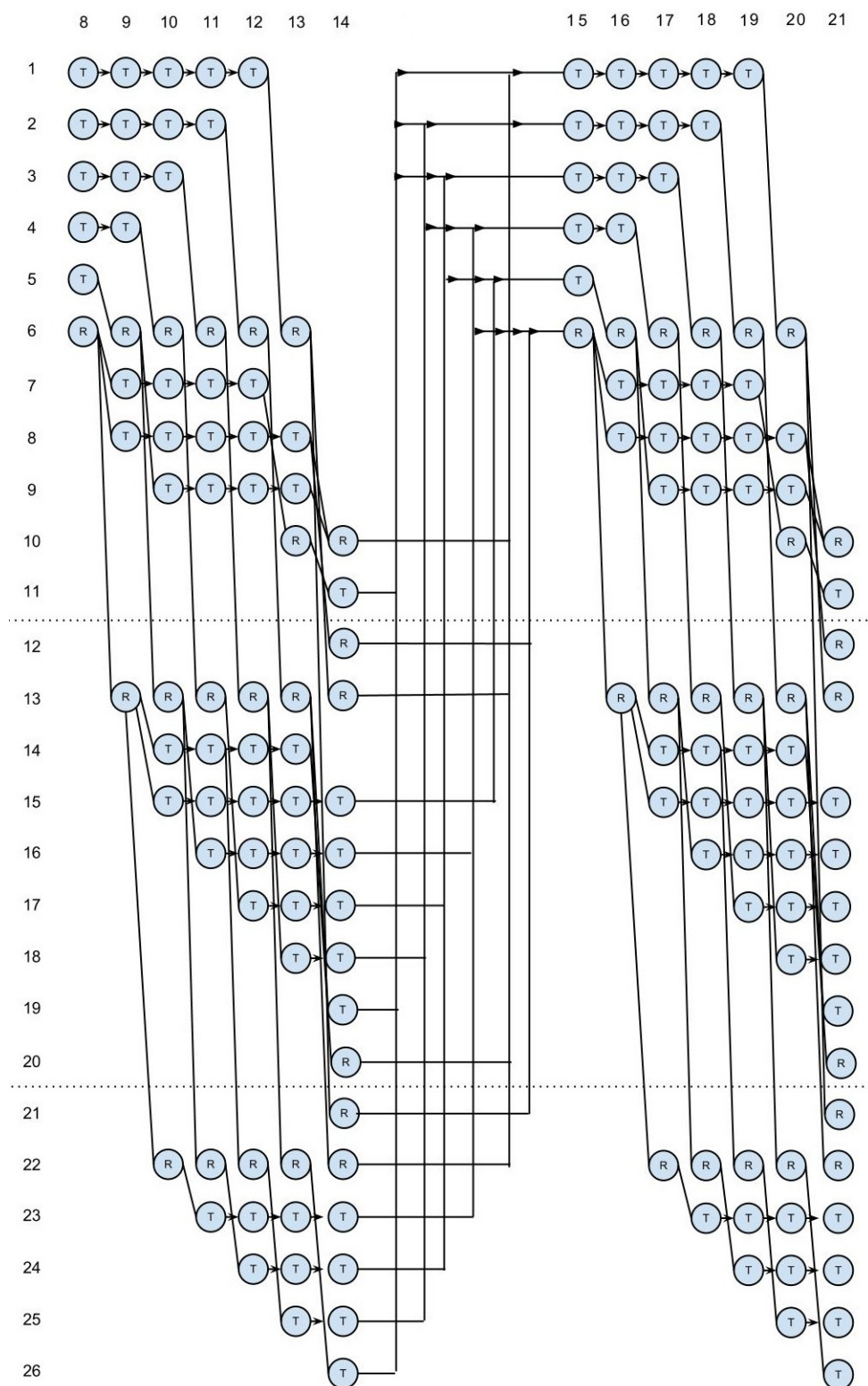


Figure 4.3 Modèle de flot

Au cours de cette section, nous allons voir comment il est possible de modéliser la plupart des contraintes. La contrainte sur le respect des journées de travail, les contraintes sur les nombres minimaux de configuration des repos et les contraintes sur le nombre minimum de repos par groupe d'employés seront présentées dans la section 4.3.

4.2.1 Bornes sur le nombre de repos par semaine

Dans la figure 4.3, les nœuds R représentent les journées de repos et les nœuds T représentent les journées de travail. La première règle modélisée dans ce flot est le minimum et le maximum de jours de repos par semaine. Dans la figure, la contrainte modélisée est un minimum de 2 jours de repos et un maximum de 3 jours de repos. Dans la première section de la figure, nous avons une première ligne de repos à la ligne 6. Cette ligne représente un premier repos dans la semaine. La deuxième ligne de repos à la ligne 10 représente un deuxième repos dans la semaine. À partir du moment où le flot passe par cette deuxième ligne, il peut se rendre à la deuxième semaine. Avant d'avoir passé cette ligne de repos, il était impossible de se rendre à la deuxième semaine. Il est donc impossible d'avoir moins de 2 repos comme il est spécifié dans la règle. Dans chacune des trois sections de la figure, les nœuds sont organisés de façon à ce qu'il soit impossible d'avoir plus de trois repos dans la semaine. Dans la première section, il n'y a que deux lignes de repos. Il aurait pu y en avoir trois, mais à cause d'une autre contrainte que nous verrons un peu plus tard, il n'y a que deux lignes de repos. Dans la deuxième section, il n'y a que deux lignes de repos, car un flot qui arrive dans la deuxième section a déjà eu un repos à la première section. Dans la troisième section, il n'y a qu'une ligne de repos, car un flot qui arrive à cette section a déjà eu un repos à la première et à la deuxième section. Le minimum et le maximum doivent donc être respectés. Il serait possible d'ajouter et d'enlever des lignes de repos si le maximum de journée de repos changeait. Il serait aussi possible de changer les arcs vers la seconde semaine pour changer le minimum de journées de repos par semaine.

4.2.2 Bornes sur les séquences de travail

La deuxième contrainte est le nombre minimum et le nombre maximum de jours de travail consécutifs. Dans l'exemple de la figure, le minimum de jours de travail consécutifs est de 4 et le maximum est de 6. À la ligne 7, lorsque le flot arrive à cette ligne, il vient d'avoir un repos de la ligne 6, colonne 1. Il lui est impossible de terminer sa séquence de travail sans qu'il ait travaillé 4 jours, comme le minimum l'oblige. Après avoir travaillé 4 jours, le flot se rend obligatoirement à un repos. S'il était plus rentable de faire travailler 5 jours, le flot pourrait se diriger vers la ligne 8. Dans ce cas, il y aurait une séquence de travail d'une durée de 5

jours suivis d'un repos. En commençant la semaine avec un repos, il n'est pas possible d'avoir 6 jours de travail consécutifs dans la même semaine. En effet, puisqu'on veut au minimum 2 jours de repos, il faut au maximum 5 jours de travail par semaine. Par contre, il est possible d'avoir 6 jours de travail consécutifs entre deux semaines. Pour empêcher cela, on contrôle les liens entre les deux semaines. Par exemple, si le flot choisit la ligne 17 pour finir la semaine 1, il peut être lié à la semaine 2 par la ligne 3, 4 ou 5. Dans chacun des cas, le minimum et le maximum de jours de travail consécutifs sont respectés. Les arcs entre les deux semaines sont établis de sorte que cette règle soit toujours respectée.

Pour augmenter ou diminuer le nombre minimum de jours de travail, il faut ajouter ou enlever des lignes entre les jours de repos. Par exemple, si le minimum est réduit à 3, il faudrait ajouter quelques lignes entre la ligne 6 et la ligne 7 permettant de commencer une séquence de travail de 3 jours à la colonne 2, 3 ou 4. Il faut aussi ajouter des liens entre la semaine 1 et la semaine 2. Pour augmenter ou diminuer le nombre maximum de jours de travail, il faut ajouter ou enlever des liens entre la semaine 1 et la semaine 2. Par exemple, pour augmenter le nombre maximum de jours de travail consécutifs à 7, il faudrait ajouter un lien entre la ligne 16 de la semaine 1 et la ligne 3 de la semaine 2.

4.2.3 Minimum de repos consécutifs

La contrainte sur le nombre minimum de deux repos consécutifs n'est pas modélisée dans le flot de la figure 4.3. Par contre, il est facile d'ajouter cette contrainte. Il suffit de retirer tous les liens entre la première semaine et la deuxième semaine qui partent de la première section en pointillé. Pour ajouter la contrainte qu'il doit y avoir au moins trois repos consécutifs, il suffit de conserver seulement les liens partant de la troisième section de la première semaine.

4.2.4 Jours de repos forcé

La contrainte des jours de repos forcés n'est pas modélisée dans le flot de la figure 4.3. Pour ajouter la contrainte à ce flot, il suffit d'éliminer toutes les lignes de travail ayant une journée de travail le jour de repos forcé. Ainsi, il sera impossible de travailler le jour du repos forcé. Par exemple, pour obliger le repos le dimanche pour le groupe d'employés, il suffit d'enlever les lignes 11, 15 à 19 et 23 à 26. Toutes ces lignes contiennent une journée de travail à la 14e colonne qui représente le dimanche.

4.2.5 Groupes d'employés et cycle

Chaque groupe d'employés a des règles et des contraintes différentes. Le modèle de flot représente un ensemble de règles. Les règles peuvent être par semaine ou par cycle. Le modèle de flot représente un cycle. Le nombre de semaines dans le flot correspond à la taille de l'horaire multipliée par le nombre de positions qu'il y a dans le cycle. Chaque unité de flot entrant dans ce modèle représente un cycle. Pour reconstruire les positions d'un cycle, il suffit de prendre une unité de flot et de la suivre dans le modèle. À chaque fois qu'on parcourt un nombre de nœuds égal à la taille de l'horaire multiplié par 7 jours, on change de position dans le cycle. Si, à un branchement, il y a plusieurs flots partant dans plusieurs directions, il suffit de continuer avec une des branches. Peu importe le choix qui sera fait, cela ne fera aucune différence. Il y a autant de flots qui entrent et qui sortent d'un nœud. Il sera toujours possible de prendre l'autre embranchement à la construction suivante. De plus, comme le coût a déjà été considéré lorsque le flot a été choisi, le fait de prendre une branche avant l'autre ne changera pas le coût total de la solution.

Étant donné que les horaires sont cycliques, il faut que le début et la fin soient compatibles. C'est-à-dire qu'un employé finissant un cycle doit pouvoir commencer le même cycle sans avoir enfreint une règle. Par exemple, un employé finissant par 2 jours de travail doit commencer le cycle avec au minimum 2 jours de travail et maximum 4 jours de travail pour respecter la contrainte sur les jours de travail consécutifs. Pour répondre à ce besoin, reprenons la figure 4.2 et modifions-la légèrement. Prenons le même modèle à 4 semaines et copions-le autant de fois qu'il y a de façons de commencer une semaine. Dans notre cas, il est possible de commencer les semaines de 6 façons différentes. Il est possible de commencer la semaine avec une journée de repos. Il est aussi possible de commencer avec 1 jour de travail. Il est possible d'avoir 2, 3, 4 ou 5 jours de travail. C'est ce que la figure 4.4 représente. Chaque

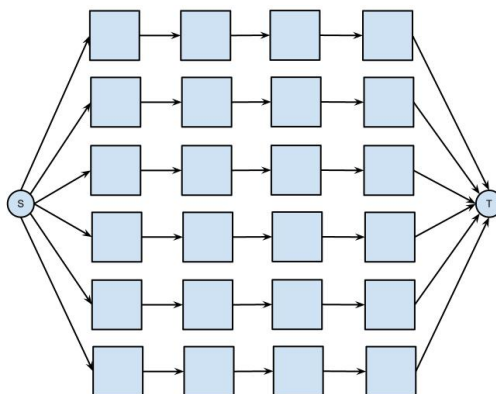


Figure 4.4 Modèle de flot à 4 semaines, avec multiples commencements

ligne représente une façon différente de commencer le cycle et chaque ligne dicte comment il est possible de finir le cycle pour qu'il soit compatible avec le début du cycle. Les seuls arcs se rendant au puits sont les arcs compatibles avec le seul arc provenant de la source de la même ligne. C'est ainsi qu'on s'assure que la fin d'un cycle est compatible avec son début.

4.2.6 Bornes sur le nombre de positions par groupe

La contrainte sur le nombre minimum et le nombre maximum de positions par groupe est facilement contrôlable. Dans la figure 4.5, un nœud a été ajouté entre la source et les différents flots représentant un groupe d'employés. Entre la source et le nouveau nœud, on limite le flot avec une borne inférieure équivalente au nombre minimum de positions dans le groupe d'employés divisé par le nombre de positions dans les cycles de ce groupe. Sur le même arc, une borne supérieure sera ajoutée. Cette borne sera l'équivalent du nombre maximum de positions dans le groupe d'employés divisé par le nombre de positions dans les cycles du groupe. Puisqu'il s'agit d'un modèle de flot, il faut s'assurer que la borne soit entière. Pour la borne minimale, on prend l'arrondi supérieur. Pour la borne supérieure, on prend l'arrondi inférieur.

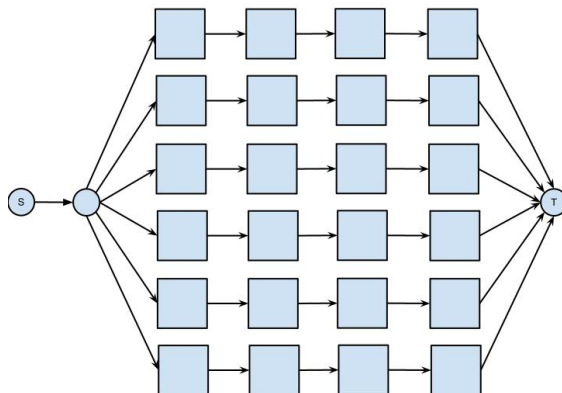


Figure 4.5 Modèle de flot à 4 semaines, avec multiples commencements et bornes sur le nombre de positions

4.2.7 Bornes sur le nombre de repos par position

Il pourrait y avoir une contrainte sur le nombre minimum et le nombre maximum de repos par position. Considérons maintenant que toutes les semaines doivent avoir au moins 1 repos. Cette considération permettra de faciliter la représentation de la contrainte. De plus, il est probable que toutes les compagnies de transport public obligent un minimum d'un repos par semaine. Considérons maintenant que tous les employés travaillent au moins une journée

par semaine. Sur 4 semaines, le minimum de repos est de 4 et le maximum est de 24. Pour modéliser cette contrainte, reprenons la figure 4.2. Cette figure représente 4 semaines. Il est possible de compter le nombre de repos total dans une position en comptant le nombre de repos pour chaque semaine. Dans la figure 4.6, chaque carré représente une semaine. Chaque semaine est liée avec la semaine suivante sur la même ligne et les semaines sur les 5 lignes en dessous. Les arcs liant une semaine à une autre représentent le nombre de repos de la première semaine. Un flot passant d'une semaine à l'autre sur la même ligne représente un flot avec un seul repos la première semaine. Un flot passant d'une semaine sur une ligne à une ligne en dessous est un flot avec deux repos la première semaine. Au total, il y a six arcs entre chaque semaine. À la quatrième colonne, la ligne représente le nombre de repos qu'il y a eu au total durant les 3 premières semaines. Pour imposer la contrainte sur le minimum et le maximum de repos, il suffit de lier les nœuds de la quatrième colonne avec le puits qui permettent d'avoir le minimum et le maximum de repos par positions.

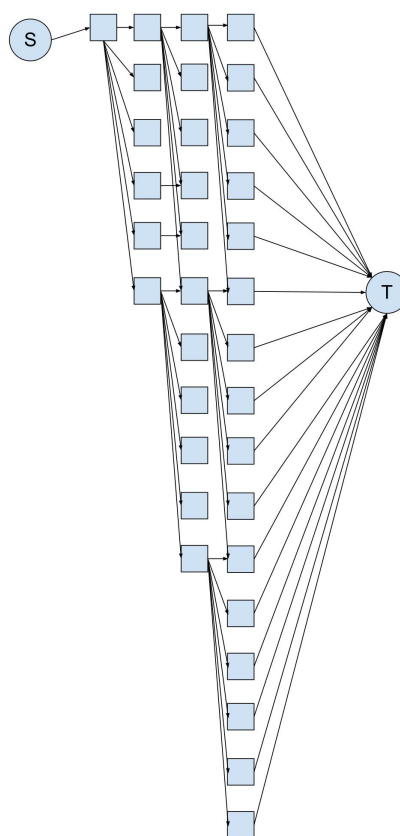


Figure 4.6 Modèle de flot à 4 semaines, borne sur le nombre de repos par position

4.3 Flots secondaires

Quelques contraintes ne peuvent pas directement être modélisées dans le flot présenté dans la section précédente. Par contre, elles peuvent être représentées dans un second flot. Avant d'expliquer comment modéliser les dernières contraintes dans un second flot, un concept doit être présenté.

Supposons qu'une copie d'un nœud peut se retrouver dans deux flots complètement séparés. Nous appellerons ceux-ci des nœuds parallèles. La contrainte qui lie ces nœuds parallèles est que la quantité de flots passant par un nœud doit être identique à la quantité passant par le nœud parallèle. Dans la figure 4.7, deux nœuds parallèles sont présentés. Le premier a des arcs pointillés représentant les arcs du premier flot. Le deuxième a des arcs pleins représentant les arcs du deuxième flot. Le troisième nœud représente les deux nœuds parallèles combinés avec les différents types d'arc. Nous utiliserons ces nœuds parallèles dans les seconds flots servant à modéliser les contraintes restantes.

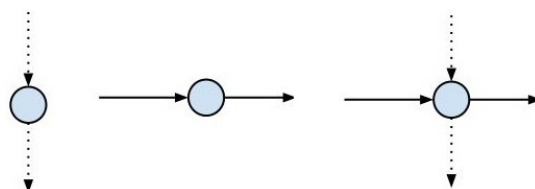


Figure 4.7 Combinaison de deux flots avec des nœuds parallèles

4.3.1 Couverture des tâches

À la section 3.4.2, un premier modèle de flot a été présenté. Il servait à s'assurer qu'il y ait assez de jours de travail pour affecter toutes les tâches. Ce flot peut être repris et lié au flot des contraintes. Pour lier ces deux flots, des nœuds parallèles seront utilisés. Pour faire cela, prenons par exemple les 5 nœuds de travail de la colonne 1 dans la figure 4.3. Ces nœuds doivent se partager les tâches à affecter. Pour lier le deuxième flot de la figure 3.1 aux 5 nœuds de travail, on intègre une copie des nœuds de travail dans le deuxième flot. Ce qui est représenté par la figure 4.8. Dans cette figure, le nœud puits est représenté par le nœud P. Les nœuds de travail parallèle provenant du premier flot sont représentés par la lettre T. Puisque les flots des tâches doivent obligatoirement être égaux à 1, ils doivent se rendre aux nœuds de travail de l'un des groupes d'employés. Les arcs du modèle de distribution des tâches sont représentés en pointillé. Les arcs du modèle des contraintes sont représentés en gris pâle. Puisque la quantité de flots passant par les nœuds dans le premier modèle doit être

égale à la quantité de flots passant par le deuxième modèle, on s'assure qu'il y aura assez d'employés travaillant la journée pour couvrir toutes les tâches à affecter.

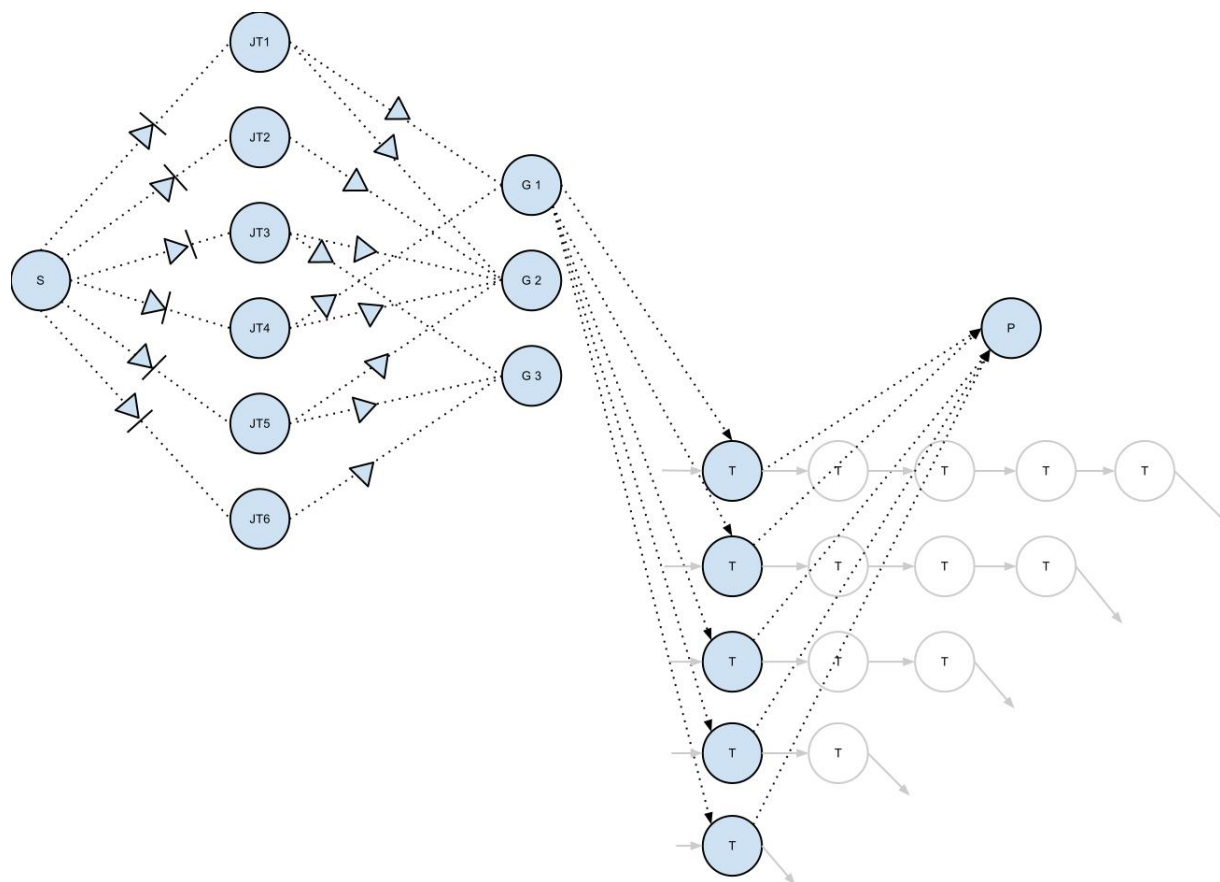


Figure 4.8 Couverture des tâches avec plusieurs flots

4.3.2 Minimum de configuration de repos

Un autre modèle de flot peut être utilisé pour s'assurer d'avoir un minimum de fin de semaine de repos, un minimum de double repos et un minimum de triple repos par semaine par groupe d'employés. Reprenons la figure 4.3. Nous avons séparé le flot en trois sections. Une section pour les simples repos, une section pour les doubles repos et une section pour les triples repos. Pour avoir assez de double repos, il faut s'assurer qu'il y ait un minimum de flot sortant de la section des doubles repos. Pour faire cela, un autre flot parallèle sera utilisé. Reprenons les nœuds de la colonne 14 ayant un arc sortant de la section des doubles repos. Ces nœuds se retrouvent dans un deuxième flot où le minimum de double repos est considéré. La figure 4.9 montre ce deuxième flot. La contrainte du minimum de double repos se trouve sur l'arc partant de la source et allant au nœud M. Le même principe peut être

appliqué pour les triples repos et sur les repos de fin de semaine.

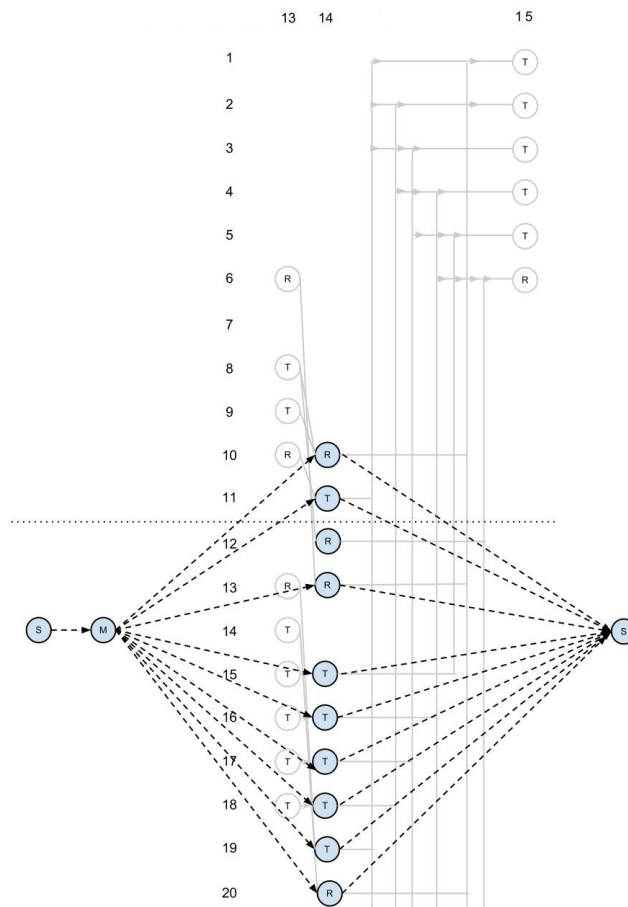


Figure 4.9 Minimum de configuration de repos avec plusieurs flots

4.3.3 Bornes sur le nombre de repos par groupe d'employés

Un autre modèle de flot peut être utilisé pour s'assurer d'avoir un minimum et un maximum de jour de repos par groupe d'employés. Les nœuds de repos de la figure 4.3 seront réutilisés dans un deuxième modèle de flot qui représentera les contraintes sur le minimum et le maximum de repos par groupe d'employés. La figure 4.10 montre une partie de ce deuxième flot. Les contraintes de minimum et de maximum sont mises comme borne inférieure et supérieure sur l'arc entre le nœud de source « S » et le nœud « M ». Dans le flot complet, tous les nœuds de repos de toutes les semaines sont liés au nœud « M » et au nœud de puits « P ».

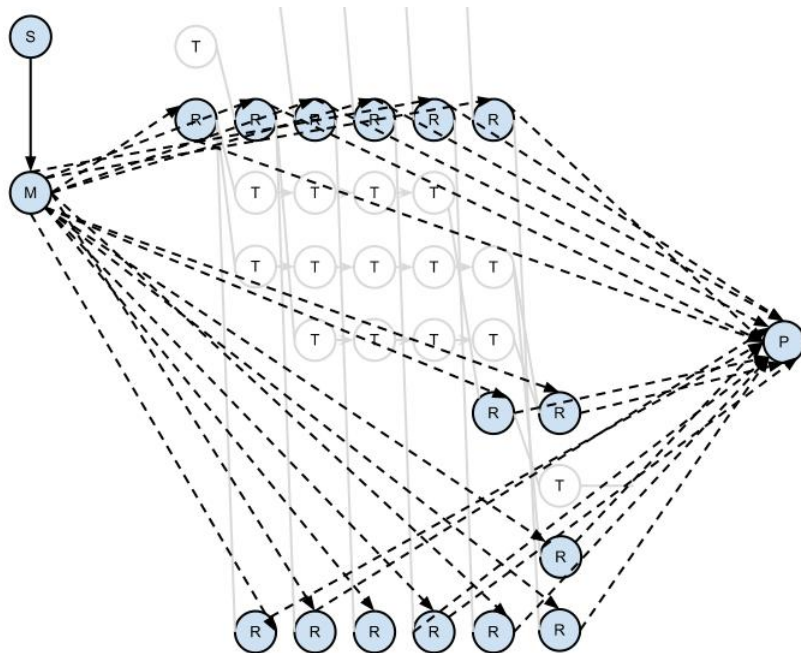


Figure 4.10 Bornes sur le nombre de repos par groupe d'employés

4.4 Méthode de résolution

4.4.1 Méthode linéaire

La combinaison des flots parallèles tels que décrits dans la section précédente élimine une partie des caractéristiques des flots. Elle ne permet pas à un algorithme de flot de résoudre le modèle de flots. Les contraintes liantes entre les flots ne peuvent pas en effet être modélisées dans un algorithme de flot.

Il est possible de modéliser un flot standard en un modèle linéaire. La quantité de flots passant sur les arcs devient une variable en nombre réel et les nœuds deviennent des contraintes de conservation de flot. La caractéristique de ces modèles linéaires est qu'ils vont trouver une solution optimale en nombre entier, car les contraintes sont toutes en nombre entier. Si on met le problème de plusieurs flots dans un solveur linéaire, on ne peut pas transformer toutes les variables représentant les arcs en variables en nombres réels. Les contraintes liantes entre deux flots éliminent la propriété que la solution optimal peut-être déterminée en relaxant les contraintes d'intégrité des variables.

Pour s'assurer qu'on obtient un résultat en nombre entier, on doit donc transformer des variables en nombre entier. Le modèle avec des nombres entiers permettra à un solveur li-

néaire en nombre entier de résoudre le modèle. En général, dans ce genre de solveur, plus il y a de variables entières, plus le temps de résolution est long. Il n'est cependant pas essentiel de transformer toutes les variables en nombre entier. On peut transformer les variables représentant la quantité de flots des arcs de quelques modèles de flots en nombre entier. Les arcs des autres flots sont en nombre réel. Il faut simplement s'assurer que pour deux flots ayant des contraintes liantes, il y a au moins un des deux flots avec des variables en nombre entier.

Cette méthode de résolution sera utilisée pour résoudre les différents problèmes dans les expérimentations. Dans ces expérimentations, le modèle de contraintes associé à chaque groupe d'employés aura ses variables en nombre réel. Tous les autres flots seront en nombre entier.

4.5 Complexité

Les contraintes sont directement déduites des règles provenant des conventions collectives et des préférences des employés. Le modèle de flot utilisé pour un groupe d'employés dépend directement des contraintes. Le modèle aura une complexité proportionnelle au nombre d'arcs et au nombre de nœuds du modèle de flot. Bref, la complexité dépend directement des règles des employés. Par contre, dans les outils de résolution actuels, la complexité ne dépend aucunement du nombre d'employés. En effet, puisque le nombre d'employés dépend du nombre de positions. Le nombre de positions dépend de la quantité de flots entrant dans chaque modèle de flot de groupe d'employé. La complexité dans un modèle de flot ne dépend aucunement de la quantité de flots entrant dans le modèle. Alors, le modèle de flot permet de gérer n'importe quelle taille d'employés sans augmentation de la complexité.

4.6 Discussion sur l'activation des positions

Dans la dernière section du chapitre précédent, la manière d'activer des positions sans activer tout un cycle a été discutée. Avec la modélisation avec plusieurs flots, nous avons le même problème. Lorsqu'une unité de flots entre dans le modèle, elle doit parcourir toute la longueur du cycle pour pouvoir en ressortir. Alors toutes les positions du cycle sont considérées.

La solution vue dans le précédent chapitre peut être appliquée ici. Il est possible de construire des cycles plus petits dans lesquels une unité de flots parcourra un plus petit nombre de positions avant d'en ressortir. Les contraintes entre le début et la fin du flot resteront les mêmes. L'inconvénient de cette approche est que la complexité augmente beaucoup

à chaque fois qu'on ajoute un flot pour représenter un cycle plus petit. À l'extrême, cela pourrait faire en sorte que le modèle ne soit plus solvable.

Il existe une autre approche. À la place de dupliquer les flots, il est possible de créer des points de sortie alternatifs dans les flots existants. Ces points de sortie seront à l'extrémité de chaque position dans le flot. Ces points de sortie doivent être compatibles avec l'entrée, comme pour la sortie à la fin du cycle. Chacun de ces points de sortie serait relié à un nœud qui lui aurait un arc vers le puits. Cet arc aurait une borne supérieure à 1 pour s'assurer qu'un seul cycle a une longueur plus petite.

CHAPITRE 5

EXPÉRIMENTATIONS

Dans ce chapitre, les expérimentations seront décrites. Premièrement, il y aura les scénarios de données. Ces scénarios représentent les données provenant de différentes compagnies de transport public dans le monde. Ensuite, l'environnement de test sera décrit. Finalement, le résultat des tests et une analyse seront présentés.

5.1 Données

Dans cette section, les scénarios d'expérimentation utilisés seront présentés. Dans ces scénarios, chaque règle peut être stricte ou souple. Les règles strictes ne peuvent pas être violées, alors que les règles souples peuvent l'être. Une pénalité est associée à chaque règle souple. Les pénalités vont croître proportionnellement avec la déviation par rapport à la règle. Prenons par exemple la règle suivante : Minimum de deux jours de congé par semaine (Pénalité proportionnelle de 10). Le coût dans la fonction objectif sera de :

$$\text{Coût} = \begin{cases} 0 & \text{Si le nombre de jour de repos est } \geq 2. \\ 10(2 - \text{Nombre de jours de repos}) & \text{Sinon.} \end{cases}$$

La pénalité est ajoutée à la fonction objectif autant de fois qu'il y a de semaines qui enfreignent la règle. Par exemple, si deux semaines n'ont aucun jour de repos, une pénalité totale de 40 sera ajoutée à la fonction objectif. Pour une règle comme le nombre minimum de repos consécutifs par semaine (Pénalité proportionnelle de 10), dès que la règle est enfreinte pour une semaine, la pénalité est ajoutée. Par exemple, une semaine qui compte deux journées de repos non consécutives, la pénalité de 10 sera ajoutée à la fonction objectif. Si une semaine compte seulement 1 jour de repos, la pénalité de 10 est aussi ajoutée à la fonction objectif.

Voici les scénarios numérotés de P1 à P5 :

P1 : Petit scénario (taille de cycle=1, nombre de semaines=1)

Caractéristiques :

Nombre de groupes d'employés : 7

Nombre de semaines : 1

Taille des cycles pour tous les groupes d'employés : 1

Nombre de tâches par colonne :

| Dimanche | Lundi | Mardi | Mercredi | Jeudi | Vendredi | Samedi |
|----------|-------|-------|----------|-------|----------|--------|
| 65 | 124 | 124 | 124 | 124 | 124 | 90 |

Nombre de positions initial par groupe d'employés. Ce nombre de positions a été déterminé pour s'assurer qu'il y ait suffisamment de positions pour accueillir toutes les journées de travail du groupe d'employés.

| #1 | #2 | #3 | #4 | #5 | #6 | #7 |
|----|----|-----|----|----|-----|----|
| 14 | 14 | 375 | 23 | 23 | 345 | 25 |

Contraintes groupe d'employés #1

- Exactement trois repos par semaine. (Stricte)
- Minimum de deux repos consécutifs par semaine. (Stricte)
- Minimum de trois repos consécutifs par semaine. (Pénalité proportionnelle de 10)

Contraintes groupes d'employés #2 à #7

- Exactement deux repos par semaine. (Stricte)
- Minimum de deux repos consécutifs par semaine. (Stricte)

P2 : Scénario moyen (taille de cycle=1, nombre de semaines=1)

Caractéristiques :

Nombre de groupes d'employés : 14

Nombre de semaines : 1

Taille des cycles pour tous les groupes d'employés : 1

Nombre de tâches par colonne :

| Dimanche | Lundi | Mardi | Mercredi | Jeudi | Vendredi | Samedi |
|----------|-------|-------|----------|-------|----------|--------|
| 170 | 370 | 370 | 370 | 370 | 370 | 219 |

Nombre de positions initial par groupe d'employés. Ce nombre de positions a été déterminé pour s'assurer qu'il y ait suffisamment de positions pour accueillir toutes les journées de travail du groupe d'employés.

| #1 | #2 | #3 | #4 | #5 | #6 | #7 | #8 | #9 | #10 | #11 | #12 | #13 | #14 |
|----|----|-----|----|----|-----|----|----|-----|-----|-----|-----|-----|-----|
| 30 | 30 | 214 | 30 | 30 | 338 | 30 | 30 | 133 | 325 | 30 | 30 | 299 | 30 |

Contraintes pour les groupes d'employés #1 et #8

- Exactement trois repos par semaine. (Stricte)
- Minimum de deux repos consécutifs par semaine. (Stricte)
- Minimum de trois repos consécutifs par semaine. (Pénalité proportionnelle de 10)

Contraintes pour les groupes d'employés #2 à #7 et #9 à #14

- Exactement deux repos par semaine. (Stricte)
- Minimum de deux repos consécutifs par semaine. (Stricte)

P3 : Scénario moyen (taille de cycle=taille du groupe, nombre de semaine=1)

Caractéristiques :

Nombre de groupes d'employés : 5

Nombre de semaines : 1

Taille des cycles pour tous les groupes d'employés : taille des cycles = taille des groupes d'employés

Nombre de tâches par colonne :

| Dimanche | Lundi | Mardi | Mercredi | Jeudi | Vendredi | Samedi |
|----------|-------|-------|----------|-------|----------|--------|
| 26 | 43 | 48 | 43 | 43 | 43 | 27 |

Nombre de positions dans chaque groupe d'employés :

| #1 | #2 | #3 | #4 | #5 |
|----|----|----|----|----|
| 13 | 13 | 13 | 13 | 9 |

La taille des groupes d'employés est fixe. Toutes les positions doivent être actives.

Contraintes pour les groupes d'employés #1 à #4

- Minimum de deux repos consécutifs par semaine. (Stricte)
- Maximum de quatre jours de repos par semaine. (Stricte)
- Exactement 31 jours de repos par groupe d'employés. (Pénalité proportionnelle de 250)
- Maximum de 5 jours de travail consécutifs. (Pénalité proportionnelle de 70)
- Maximum de 6 jours de travail consécutifs. (Pénalité proportionnelle de 70)
- Maximum de 7 jours de travail consécutif. (Stricte)

Contraintes pour le groupe d'employés #5

- Minimum de deux repos consécutifs par semaine. (Stricte)
- Maximum de quatre jours de repos par semaine. (Stricte)
- Exactement 21 jours de repos par groupe d'employés. (Pénalité proportionnelle de 250)
- Maximum de 5 jours de travail consécutifs. (Pénalité proportionnelle de 70)
- Maximum de 6 jours de travail consécutifs. (Pénalité proportionnelle de 70)
- Maximum de 7 jours de travail consécutifs. (Stricte)

P4 : Scénario moyen (taille de cycle=1,2 ou 4, nombre de semaines=2)

Caractéristiques :

Nombre de groupes d'employés : 4

Nombre de semaines : 2

Taille des cycle du groupe d'employés #1 : 1

Taille des cycle du groupe d'employés #2 : 2

Taille des cycle du groupe d'employés #3 : 4

Taille des cycle du groupe d'employés #4 : 1

Nombre de tâches par colonne :

| Dim. | Lun. | Mar. | Mar. | Jeu. | Ven. | Sam. | Dim. | Lun. | Mar. | Mar. | Jeu. | Ven. | Sam. |
|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| 100 | 200 | 200 | 200 | 200 | 200 | 125 | 100 | 200 | 200 | 200 | 200 | 200 | 125 |

Nombre de positions initial par groupe d'employés. Ce nombre de positions a été déterminé pour s'assurer qu'il y ait suffisamment de positions pour accueillir toutes les journées de travail du groupe d'employés.

| #1 | #2 | #3 | #4 |
|-----|-----|-----|----|
| 100 | 200 | 100 | 10 |

Contraintes pour le groupe d'employés #1

- Tous les dimanches doivent être un repos (Pénalité proportionnelle de 50)
- Exactement 2 repos consécutifs par semaine par position. (Stricte)
- Pour chaque semaine, minimum de 20 positions avec le samedi et le dimanche de repos. (Stricte)
- Maximum de 6 jours consécutifs de travail. (Stricte)

Contraintes pour le groupe d'employés #2

- Exactement 3 jours de repos par semaine par position. (Stricte)
- Pour chaque semaine, au moins 20 positions avec au moins 2 repos consécutifs. (Stricte)
- Pour chaque semaine, au moins 10 positions avec au moins 3 repos consécutifs. (Stricte)
- Maximum de 6 jours consécutifs de travail. (Stricte)

Contraintes pour le groupe d'employés #3

- Minimum de 20 positions actives. (Stricte)
- Minimum de 4 jours de travail consécutifs. (Stricte)
- Maximum de 6 jours de travail consécutifs. (Stricte)
- Maximum de 5 jours de travail consécutifs. (Pénalité proportionnelle de 50)
- Minimum de deux jours consécutifs de repos. (Pénalité proportionnelle de 50)

Contraintes pour le groupe d'employés #4 (Temps partiel)

- Minimum de 4 jours de repos par semaine. (Stricte)
- Maximum de 3 jours de travail consécutifs. (Stricte)

P5 : Gros scénario (taille de cycle=taille du groupe, nombre de semaines=4)

Caractéristiques :

Nombre de groupes d'employés : 19

Nombre de semaines : 4

Taille des cycles pour les groupes d'employés de #1 à #18 : taille des cycles = taille des groupes d'employés

Taille des cycles du groupe d'employés #19 : 1

Nombre de tâches par colonne :

| Dim. | Lun. | Mar. | Mar. | Jeu. | Ven. | Sam. | Dim. | Lun. | Mar. | Mar. | Jeu. | Ven. | Sam. |
|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| 66 | 253 | 253 | 253 | 253 | 253 | 115 | 66 | 253 | 253 | 253 | 253 | 253 | 115 |
| Dim. | Lun. | Mar. | Mar. | Jeu. | Ven. | Sam. | Dim. | Lun. | Mar. | Mar. | Jeu. | Ven. | Sam. |
| 66 | 253 | 253 | 253 | 253 | 253 | 115 | 66 | 253 | 253 | 253 | 253 | 253 | 115 |

Nombre de positions par groupe d'employés :

| #1 | #2 | #3 | #4 | #5 | #6 | #7 | #8 | #9 | #10 |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 6 | 32 | 10 | 8 | 14 | 8 | 17 | 25 | 90 | 10 |
| #11 | #12 | #13 | #14 | #15 | #16 | #17 | #18 | #19 | |
| 29 | 11 | 2 | 2 | 8 | 12 | 2 | 3 | 100 | |

La taille des groupes d'employés #1 à #18 est fixe. Toutes les positions doivent être actives.

La taille du groupe d'employés #19 est assez grande pour pouvoir accueillir toutes les journées de travail en surplus qui ne rentrent pas dans les autres groupes d'employés.

Contraintes pour les groupes d'employés #1, #13 et #17

- Minimum de 1 repos par semaine. (Stricte)
- Maximum de 6 jours de travail consécutifs. (Stricte)

Contraintes pour les groupes d'employés #2, #4, #5, #10, #12, #14, #15, #16 et #18

- Minimum de 2 jours de congé par semaine. (Pénalité proportionnelle de 250)
- Maximum de 3 jours de congé par semaine. (Pénalité proportionnelle de 250)
- Exactement 9 jours de congé par position. (Pénalité proportionnelle de 250)

Contraintes pour les groupes d'employés #3, #6, #7, #8, #9 et #11

- Minimum de 2 jours de congé par semaine. (Stricte)
- Maximum de 3 jours de congé par semaine. (Stricte)
- Exactement 9 jours de congé par position. (Stricte)

Contraintes pour les groupe d'employés #19

- Minimum de 2 jours de congé par semaine. (Stricte)
- Maximum de 3 jours de congé par semaine. (Stricte)
- Chaque position active a un coût de 100 dans la fonction objectif.

5.2 Environnement de test

Tous les tests ont été exécutés sur le même ordinateur. Celui-ci a les caractéristiques suivantes :

Ordinateur Intel Core 2 Duo E8500 (Dual core de 3.16Ghz), 8GO de RAM

System d'exploitation : Windows 64bits

Cplex : 12.2.0.0

IBM ILOG CPLEX Optimization Studio : 12.2

5.3 Résultats

5.3.1 Modélisation linéaire

Voici les résultats de l'expérimentation de l'implémentation de la modélisation linéaire décrite dans le chapitre 3. En ordre, les informations présentées dans le tableau sont : l'identificateur du scénario de données, une valeur indiquant si la solution obtenue à la fin a un coût optimal, le temps d'exécution de CPLEX, le coût de la solution obtenue, le nombre de colonnes du modèle linéaire fournis à CPLEX, le nombre de lignes du modèle linéaire fournis à CPLEX, le nombre de colonnes après l'étape du MIP Presolve de CPLEX, le nombre de lignes après l'étape du MIP Presolve de CPLEX, le gap d'intégralité entre la racine de l'arbre de branchement et la solution obtenue.

| Scénario : | P1 | P2 | P3 | P4 | P5 |
|---|-----------|-----------|-----------|-----------|-----------|
| Optimal : | Oui | Oui | Oui | Oui | Oui |
| Temps d'exécution (sec.) : | 947.63 | 10852.23 | 5.54 | 1716.09 | 46.21 |
| Valeur de l'objectif : | 0.0 | 60.0 | 170.0 | 0.0 | 1480.0 |
| Nombre de colonnes du modèle linéaire : | 27413 | 73009 | 4263 | 72204 | 231279 |
| Nombre de lignes du modèle linéaire : | 41315 | 99127 | 9253 | 192258 | 353625 |
| Nombre de colonnes après MIP Presolve : | 20629 | 40004 | 3441 | 34003 | 37824 |
| Nombre de lignes après MIP Presolve : | 28757 | 55559 | 3240 | 48233 | 54329 |
| Gap d'intégralité : | 0.0 | 60.0 | 0.0 | 0.0 | 0.0 |

5.3.2 Modélisation par les flots

Voici les résultats de l'expérimentation de l'implémentation de la modélisation par flots décrite dans le chapitre 4. Cette modélisation a été résolue à l'aide de la méthode présentée à la section 4.4.1. En ordre, les informations présentées dans le tableau sont : l'identificateur

du scénario de données, une valeur indiquant si la solution obtenue à la fin a un coût optimal, le temps d'exécution de CPLEX, le coût de la solution obtenue, le nombre de colonnes du modèle linéaire fournis à CPLEX, le nombre de lignes du modèle linéaire fournis à CPLEX, le nombre de colonnes après l'étape du MIP Presolve de CPLEX, le nombre de lignes après l'étape du MIP Presolve de CPLEX, le gap d'intégralité entre la racine de l'arbre de branchement et la solution obtenue.

| Scénario : | P1 | P2 | P3 | P4 | P5 |
|---|-----------|-----------|-----------|-----------|-----------|
| Optimal : | Oui | Oui | Oui | Oui | Oui |
| Temps d'exécution (sec.) : | 0.14 | 0.06 | 166.95 | 1.17 | 21.09 |
| Valeur de l'objectif : | 0.0 | 60.0 | 170.0 | 0.0 | 1480.0 |
| Nombre de colonnes du modèle linéaire : | 2746 | 5462 | 109179 | 24013 | 84122 |
| Nombre de lignes du modèle linéaire : | 6273 | 12476 | 353765 | 100488 | 231754 |
| Nombre de colonnes après MIP Presolve : | 197 | 344 | 107334 | 18208 | 80544 |
| Nombre de lignes après MIP Presolve : | 60 | 118 | 39162 | 5291 | 29323 |
| Gap d'intégralité : | 0.0 | 7.5 | 0.0 | 0.0 | 0.0 |

5.4 Analyse

Les tableaux de la section précédente montrent les résultats des expérimentations. On peut constater que les deux modélisations permettent d'atteindre une solution optimale dans un temps raisonnable. L'expérimentation la plus longue est le scénario P2, qui a été résolu par la modélisation par variables binaires. Cette expérimentation a pris un peu plus de 3 heures.

On observe également que la modélisation par flot est généralement plus rapide que la modélisation par variables binaires. En moyenne, le temps pris dans le premier tableau est de 45,23 minutes. Dans le deuxième, la moyenne est de 37,88 secondes, ce qui est beaucoup plus rapide. Par contre, pour le scénario P3, la première modélisation est plus rapide. La première modélisation a pris 5,54 secondes pour résoudre le problème comparativement à 166,95 pour la deuxième modélisation. Cela s'explique par le fait que le problème est très contraint au niveau des tâches à distribuer. Le nombre de positions requises a été précisément déterminé par les utilisateurs qui sont habitués de construire les horaires pour cette compagnie de transport public. Il n'y a donc aucune complexité au niveau de l'activation et de la désactivation des positions. De plus, étant donné que les contraintes sont sur un cycle de 9 et de 13 semaines,

on obtient un modèle de flot plus gros que les autres scénarios.

Même si le temps de résolution est plus long pour le scénario P3, la modélisation par flot a permis de résoudre tous les problèmes en l'espace de quelques minutes. Le temps de résolution est plus constant avec cette modélisation. De plus, si le nombre de tâches et le nombre d'employés augmentent, cela n'aura aucun effet sur la taille des problèmes. Cela a l'avantage qu'une fois que l'algorithme est implémenté et utilisé par une compagnie, si le service change et qu'il y a plus de tâches, le temps de résolution sera constant. D'un autre côté, si les règles des conventions collectives changent, les deux modèles pourraient prendre plus ou moins de temps à construire l'horaire.

CHAPITRE 6

CONCLUSION

6.1 Synthèse des travaux

Le problème de création des séquences de jours de travail et de jours repos est un sujet qui a été étudié par plusieurs chercheurs. Dans la revue de littérature, nous avons vu plusieurs techniques pour résoudre ce problème. Cependant, ces techniques permettaient seulement de résoudre une partie du problème. Peu de techniques permettaient de prendre en compte les horaires cycliques. Aucune méthode n'avait été présentée pour résoudre le problème avec plusieurs groupes d'employés interdépendants.

Dans ce mémoire, nous avons vu qu'il était possible de résoudre les problèmes de détermination de la taille des effectifs de multiples groupes d'employés et d'affectation des séquences de repos dans compagnies de transport public. Nous avons vu qu'il existe au moins deux méthodes permettant de résoudre ce problème.

La première méthode présentée est un modèle linéaire où les jours sont représentés par une variable. Cette variable prend la valeur 1 si le jour est un repos, 0 sinon. Une autre variable binaire représente l'activation des positions. Une position active est une position qu'un employé fera. Les positions inactives permettent de s'assurer qu'il y aura exactement le bon nombre de positions actives pour couvrir le travail tout en respectant les règles. Ces deux types de variables binaires ont permis d'écrire toutes les contraintes qui s'appliquent au problème.

Dans la phase d'expérimentation, nous avons vu que la modélisation binaire pouvait être très rapide à résoudre, mais aussi parfois assez longue. Les résultats variaient entre 5 secondes et 3 heures. Dans tous les scénarios de données, la modélisation a réussi à déterminer une solution optimale.

La deuxième modélisation est une modélisation linéaire représentant une composition de plusieurs flots. Le premier flot représente les séquences valides dans un cycle. Le deuxième flot permet de représenter la distribution des tâches pour s'assurer d'avoir assez de jours de travail pour couvrir toutes les tâches. Les autres flots permettent de respecter des règles qui s'appliquent sur l'ensemble d'un groupe de travail.

Dans les expérimentations, nous avons vu que cette deuxième méthode est très efficace. La plupart des scénarios sont résolus plus rapidement que par la première modélisation et le temps de résolution reste plus constant avec cette deuxième méthode. Les temps de résolution varie en effet entre une fraction de seconde et 3 minutes.

Finalement, nous avons observé que la complexité de la deuxième modélisation n'est pas liée au nombre d'employés, contrairement à la première modélisation. Seules les règles et la taille des cycles ont une influence sur la complexité de la deuxième modélisation.

6.2 Limitations des solutions proposées

Une première limitation associée à la modélisation binaire est sa complexité qui grossit avec le nombre d'employés. Dans les tests que nous avons effectués, le nombre d'employés n'a jamais dépassé 450. Dans de grosses compagnies de transport public, ce nombre peut atteindre 8000 employés. La modélisation binaire pourrait avoir de la difficulté à traiter un si grand nombre d'employés. D'un autre côté, dans une compagnie à 8000 chauffeurs, des simplifications pourraient probablement être faites. Le problème pourrait peut-être être divisé sans devoir séparer les tâches dépendantes de plusieurs groupes d'employés.

La deuxième modélisation a une limitation au niveau des règles que nous pouvons combiner. Si nous voulons combiner plusieurs règles ayant des contraintes sur plusieurs semaines, la matrice de semaine telle que présentée dans la section 4.2.7 pourrait devenir très grosse. Chaque contrainte supplémentaire est associée à une valeur qui doit être calculée. Une dimension de matrice doit être ajoutée pour chaque valeur supplémentaire à calculer. Cela fait donc exploser le nombre de nœuds et d'arcs. Cependant, en pratique, le nombre de règles contraignant plusieurs semaines est généralement petit.

Dans la deuxième modélisation, si un problème contient trop d'arcs qui doivent être modélisé par une variable en nombre entier, le solveur linéaire aura de la difficulté à résoudre le problème. Plus le problème est gros, plus il sera difficile pour le solveur de bien brancher. Le nombre de groupes d'employés peut donc devenir une limitation à cette modélisation.

6.3 Améliorations futures

Pour permettre de résoudre plus facilement la deuxième modélisation, il serait possible de relaxer les contraintes liantes entre les différents modèles de flot. On pourrait utiliser une décomposition lagrangienne qui permettrait de résoudre le problème plus rapidement en

permettant de violer certaines contraintes liantes. De façon itérative, la solution convergerait peut-être vers la solution optimale.

RÉFÉRENCES

- ARABEYRE, J., FEARNLEY, J., STEIGER, F. et TEATHER, W. (1969). The airline crew scheduling problem : A survey. *Transportation Science*, 3, 140–163.
- BENNETT, B. et POTTS, R. (1968). Rotating roster for a transit system. *Transportation science*, 2, 14–34.
- BENNETT, B. T. (1967). *Optimization of bus crew rosters : an application of combinatorial mathematics*. Thèse de doctorat, University of Adelaide.
- CAPPANERA, P. et GALLO, G. (2004). A multicommodity flow approach to the crew rostering problem. *Operations Research*, 583–596.
- CAPRARA, A., TOTH, P., VIGO, D. et FISCHETTI, M. (1998). Modeling and solving the crew rostering problem. *Operations research*, 820–830.
- DAY, P. et RYAN, D. (1997). Flight attendant rostering for short-haul airline operations. *Operations research*, 649–661.
- ERNST, A., JIANG, H., KRISHNAMOORTHY, M. et SIER, D. (2004). Staff scheduling and rostering : A review of applications, methods and models. *European journal of operational research*, 153, 3–27.
- GAMACHE, M. et SOUMIS, F. (1998). A method for optimally solving the rostering problem. *Operations Research in the Airline Industry*, 124–157.
- GAMACHE, M., SOUMIS, F., MARQUIS, G. et DESROSIERS, J. (1999). A column generation approach for large-scale aircrew rostering problems. *Operations research*, 247–263.
- HARTOG, A., HUISMAN, D., ABBINK, E. et KROON, L. (2009). Decision support for crew rostering at ns. *Public Transport*, 1, 121–133.
- NURMI, K., KYNGÄS, J. et POST, G. (2011). Driver rostering for bus transit companies. *Engineering Letters*, 19, 125–132.
- SODHI, M. et NORRIS, S. (2004). A flexible, fast, and optimal modeling approach applied to crew rostering at london underground. *Annals of Operations Research*, 127, 259–281.
- STM (2011). Plan stratégique 2020, [http ://stm.info/en-bref/plan_strategique2020.pdf](http://stm.info/en-bref/plan_strategique2020.pdf).